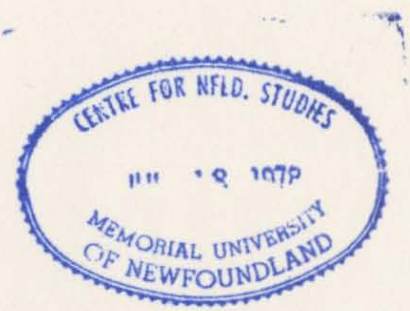
COGNITIVE VISUAL PERCEPTION MECHANISM FOR ROBOTS USING OBJECT-BASED VISUAL ATTENTION

YUANLONG YU


CENTRE FOR NFLD. STUDIES
MEMORIAL UNIVERSITY
OF NEWFOUNDLAND

Cognitive Visual Perception Mechanism for Robots Using Object-based Visual Attention

By

©Yuanlong Yu

A thesis submitted to the School of Graduate Studies
in partial fulfillment of the requirements for the degree of Doctor of Philosophy

Faculty of Engineering and Applied Science
Memorial University of Newfoundland
St. John's, Newfoundland, Canada
December, 2010

Abstract

Based on the psychological and physiological fact that humans employ a visual attention mechanism to connect perception and action by selecting the relevant parts of the environment in an unconscious or conscious way and using the relevant parts to produce an appropriate action, this thesis presents a cognitive visual perception paradigm that determines how visual inputs reach awareness and guide actions.

Based on the idea that a general way of organizing the visual scene is to parcel it into discrete objects, object-based visual attention theory is employed in the proposed paradigm. This proposed paradigm models robotic visual perception as a three-stage process: pre-attentive processing, attentional selection and post-attentive perception. It indicates that robotic visual perception starts from a low-level cognitive attentional selection procedure that guides attention to the relevant object of the scene, followed by a high-level post-attentive analysis procedure that analyzes the attended object and formulates it into an internal mental representation used for further cognitive behaviors.

The pre-attentive processing stage extracts pre-attentive features and divides the input scene into uniform proto-objects by using an irregular pyramid based segmentation method. The attentional selection stage guides attention to one proto-object of interest by means of unconscious bottom-up competition and conscious top-down biasing. The bottom-up competition is modeled by estimating the saliency of each proto-object. The top-down biasing is modeled by using integrated competition hypothesis: by directing attention to a task-relevant feature of an object, a competitive advantage over the whole object is produced. Furthermore, this thesis asserts that the task-relevant feature can be autonomously deduced from the internal representation of the task-relevant object that is specified by or inferred from the current task.

Once a proto-object is selected by attention, it proceeds to the post-attentive perception stage, which includes perceptual completion processing, extraction of post-attentive features, object recognition, and development of the internal representation of the attended object in long-term memory. The internal representation is autonomously organized and learned under the framework of probabilistic neural networks in the sense that an object is modeled as a hierarchical cluster. Thus, each instance in the cluster can be abstracted as a mental state that can be used for high-level cognitive behaviors, such as attentional prediction and action determination.

This proposed cognitive visual perception paradigm is applied into distinct robotic tasks, including detection of salient objects, detection of task-relevant objects and target tracking. Experimental results under different conditions are shown to validate this paradigm.

Acknowledgements

It is my great pleasure to finally have a chance to show my profound gratitude to peoples who accompanied me during the five years of this doctoral program.

First, I would like to express my sincerest thanks to my supervisors, Dr. George Mann and Dr. Raymond Gosine, for giving me the opportunity to carry out my doctoral research under their supervision. Special thanks go to Dr. George Mann for his valuable suggestions on my research, great help on my writings and continuous encouragement. I am grateful to another member of my supervisory committee, Dr. Peter McGuire, for his kind supervision.

I am also grateful to Memorial University of Newfoundland, Natural Science and Engineering Research Council of Canada (NSERC) and C-CORE to provide financial supports for my doctoral research.

I would like to thank all my colleagues in Intelligent System Lab at Memorial, especially Dilan Amarasinghe, Rajibul Huq, Momotaz Begum and Awantha Dewage Don. I also appreciate the help from Xiaoning Zhang and Cheng Wang in my experiments.

I would like to show my profound gratitude to my parents and my sister for their support and encouragement. I also would like to thank the support from Ms. Nan Li.

Finally, I would like to thank Dr. Fakhri Karray at University of Waterloo, Dr. Andrew Vardy and Dr. Nicholas Krouglicof at Memorial, for taking their time to read my thesis and give valuable suggestions on it.

Contents

List of Tables

List of Figures

List of Acronyms

AMD	:	Autonomous Mental Development
BC	:	Biased Competition
DDD	:	Data-Driven Decimation
EKF	:	Extended Kalman Filter
EM	:	Expectation-Maximization
FDMP	:	First-order Discrete Markov Process
FIT	:	Feature Integration Theory
FPR	:	False Positive Rate
GSM	:	Guided Search Model
HIS	:	Hue-Intensity-Saturation
IC	:	Integrated Competition
IOR	:	Inhibition of Return
IT	:	Inferior Temporal
JPDAF	:	Joint PDAF
LGN	:	Lateral Geniculate Nucleus
LTM	:	Long-Term Memory
MAP	:	Maximum A Posteriori
MLE	:	Maximum Likelihood Estimation
MT	:	Middle Temporal
PDAF	:	Probabilistic Data Association Filter
PFC	:	Prefrontal Cortex
PNN	:	Probabilistic Neural Network
PP	:	Posterior Parietal
RBF	:	Radial Basis Function

SIFT	:	Scale-Invariant Feature Transform
SLAM	:	Simultaneous Localization and Mapping
SPD	:	Stochastic Pyramid Decimation
STD	:	Standard Deviation
STEM	:	Spatio-Temporal Energy Model
TPR	:	True Positive Rate
UKF	:	Unscented Kalman Filter
WM	:	Working Memory
WTA	:	Winner-Take-All

List of Symbols

(a, b)	:	standard deviations of the Gaussian envelope of a 2-D Gabor filter
$a_i^{j,k}$	:	the occurrence number of the instance i of the part j in the local PNN of the object k
a_i^{l+1}	:	the boolean variable indicating if the vertex v_i at level l will survive at level $l+1$ during the decimation procedure of an irregular pyramid
$a_i^{l+1,n}$	:	the value of the boolean variable a_i^{l+1} at an iteration indexed by n during the decimation procedure of an irregular pyramid
act_{t+1}^{e}	:	the external action at moment t
act_{t+1}^{i}	:	the attentional prediction at moment $t+1$
$attn_t$	:	the attentional state at moment t
A_{ct}	:	the area of the closed contour represented by $\mathbf{F}_{ct}^{t,\mu}$
A_{FP}	:	the number of pixels that are in the tracked region but not in the real target
A_i^l	:	the area of a vertex v_i at level l in an irregular pyramid
A_k^{l+1}	:	the area of a vertex v_k at level $l+1$ in an irregular pyramid
A_{real}	:	the pixel number of the real target during tracking
$A_{\mathbf{R}_g}$	:	the area of the g_{th} proto-object $\mathbf{R}_g$
A_{TP}	:	the number of pixels that are both in the tracked region and in the real target
A_x	:	the image width at the base level of an irregular pyramid
A_y	:	the image height at the base level of an irregular pyramid
$b_i^{j,k}$	:	the occurrence number of the control point i along a contour instance j in the global PNN of the object k

b_i^{l+1}	:	the boolean variable indicating if *Rule 2* is not satisfied for the vertex v_i at level l during the decimation procedure
$b_i^{l+1,n}$	:	the value of the boolean variable b_i^{l+1} at an iteration indexed by n during the decimation procedure of an irregular pyramid
$B_{by}(\mathbf{r}_i)$	:	the top-down bias in terms of blue-yellow pair at location $\mathbf{r}_i$
$B_{ct}(\mathbf{r}_i)$	:	the top-down bias in terms of contour at location $\mathbf{r}_i$
$B_{int}(\mathbf{r}_i)$	:	the top-down bias in terms of intensity at location $\mathbf{r}_i$
$B_{mv}(\mathbf{r}_i)$	:	the top-down bias in terms of motion at location $\mathbf{r}_i$
$B_{o_\theta}(\mathbf{r}_i)$	:	the top-down bias in terms of orientation in θ at location $\mathbf{r}_i$
$B_{rg}(\mathbf{r}_i)$	:	the top-down bias in terms of red-green pair at location $\mathbf{r}_i$
$B_{tot}(\mathbf{r}_i)$	:	the total top-down bias in terms of all task-relevant feature dimensions at location $\mathbf{r}_i$
$\mathbf{b}$	:	blue channel in the Red-Green-Blue color model
$\mathbf{B}$	:	blue channel of the broadly-tuned color channels
$\mathbf{B}(z)$	:	a B-Spline basis function vector whose entries are polynomials in a real variable z
c	:	the parameter that determines the shape of the weighting function $w(m)$ or $w(m,n)$
(c_{ct}^x, c_{ct}^y)	:	centroid coordinates of the contour represented by the attentional template $\mathbf{F}_{ct}^{t,\mu}$
$(c_{\mathbf{R}_g}^x, c_{\mathbf{R}_g}^y)$	:	centroid coordinates of the g_{th} proto-object $\mathbf{R}_g$
C_{FPR}	:	the false positive rate used to evaluate target completion in the tracking task
C_{TPR}	:	the true positive rate used to evaluate target completion in the tracking task
$\mathbf{C}$	:	a contour curve
$\mathbf{C}_{\mathbf{R}_g}^m$	:	a predicted contour (indexed by m) for a proto-object (indexed by g) used to estimate the top-down bias in terms of contour

d	:	a dimension or the dimension number of a local post-attentive feature or of a global post-attentive feature
d_t	:	the measure of the temporal derivative
d_x	:	the measure of the spatial derivative in x-axis
d_y	:	the measure of the spatial derivative in y-axis
D_B	:	Bhattacharyya distance
$\mathbf{d}_s$	:	the vector of measures of spatial derivatives
$\mathbf{D}$	:	the derivative operator
e^l	:	an intra-level edge at level l in an irregular pyramid
$e^0_{i,j}$	:	the intra-level edge between vertices v_i and v_j at the base level $l = 0$ in an irregular pyramid
$e^l_{i,j}$	:	the intra-level edge between vertices v_i and v_j at level l in an irregular pyramid
$e^{l+1}_{k,k'}$	:	a candidate intra-level edge between vertices v_k and $v_{k'}$ at level $l+1$ in an irregular pyramid
$e_{\mathbf{r}_i}$	:	the random event of the location $\mathbf{r}_i$ being attended
$e_{\mathbf{R}_g}$	:	the random event of the proto-object $\mathbf{R}_g$ being attended
$\tilde{e}^l_i$	:	the sum of strength of intra-level edges of a vertex v_i at level l
$\tilde{e}^l_j$	:	the sum of strength of intra-level edges of a vertex v_j at level l
$\mathbf{E}_l$	:	the edge set in the graph $\mathbf{G}_l$
$\mathbf{E}_{l+1}$	:	the edge set in the graph $\mathbf{G}_{l+1}$
f	:	a feature dimension, e.g., intensity, red-green pair, blue-yellow pair, orientation energy in four preferred orientations, contour and motion energy
f_c	:	B-Spline basis functions
f_{rel}	:	the task-relevant feature dimension
$\{f_{rel}\}$	:	the set of task-relevant feature dimensions
$F_{ao}(\mathbf{r}^j_a)$	:	the absolute orientation of a pixel $\mathbf{r}^j_a$

$F_{ro}(\mathbf{r}_a^j)$	:	the relative orientation of a pixel $\mathbf{r}_a^j$
FP	:	the number of false positives
$F(\mathbf{r}_i)$	:	a pre-attentive feature (scalar) at the pixel $\mathbf{r}_i$
$F(\mathbf{r}_i, l)$	:	a pre-attentive feature (scalar) at the pixel $\mathbf{r}_i$ at level l
$F_{by}(\mathbf{r}_i, l)$	:	the pre-attentive feature (scalar) in terms of blue-yellow pair at the pixel $\mathbf{r}_i$ at level l
$F_{ct}(\mathbf{r}_i, l)$	:	the pre-attentive feature (scalar) in terms of contour at the pixel $\mathbf{r}_i$ at level l
$F_{int}(\mathbf{r}_i, l)$	:	the pre-attentive feature (scalar) in terms of intensity at the pixel $\mathbf{r}_i$ at level l
$F_{mv}(\mathbf{r}_i, l)$	:	the pre-attentive feature (scalar) in terms of motion energy at the pixel $\mathbf{r}_i$ at level l
$F_{o_\theta}(\mathbf{r}_i, l)$	:	the pre-attentive feature (scalar) in terms of orientation energy in orientation θ at the pixel $\mathbf{r}_i$ at level l
$F_{rg}(\mathbf{r}_i, l)$	:	the pre-attentive feature (scalar) in terms of red-green pair at the pixel $\mathbf{r}_i$ at level l
$\hat{F}_{by,i}^l$	:	the aggregate feature (scalar) in terms of blue-yellow pair of a vertex v_i at level l in an irregular pyramid
$\hat{F}_{int,i}^l$	:	the aggregate feature (scalar) in terms of intensity of a vertex v_i at level l in an irregular pyramid
$\hat{F}_{rg,i}^l$	:	the aggregate feature (scalar) in terms of red-green pair of a vertex v_i at level l in an irregular pyramid
$F_{by}^{t,\mu}$	:	the mean in the attentional template in terms of blue-yellow pair
$F_{by}^{t,\sigma}$	:	the STD in the attentional template in terms of blue-yellow pair
$F_{int}^{t,\mu}$	:	the mean in the attentional template in terms of intensity
$F_{int}^{t,\sigma}$	:	the STD in the attentional template in terms of intensity
F_{mv}^t	:	the attentional template in terms of motion
F_o^t	:	the attentional template in terms of local orientations

$F_{rg}^{t,\mu}$	:	the mean in the attentional template in terms of red-green pair
$F_{rg}^{t,\sigma}$	:	the STD in the attentional template in terms of red-green pair
$\tilde{F}_{gb,d}$	:	the global post-attentive feature in terms of the dimension d in $\tilde{\mathbf{F}}_{gb}$
$\tilde{F}_{lc,d}$	:	the local post-attentive feature in terms of the dimension d in $\tilde{\mathbf{F}}_{lc}$
$\tilde{F}_{by}^{s}(\mathbf{R}_j^{attn})$	:	the salience component in terms of blue-yellow pair of a local post-attentive feature of $\mathbf{R}_j^{attn}$
$\tilde{F}_{int}^{s}(\mathbf{R}_j^{attn})$	:	the salience component in terms of intensity of a local post-attentive feature of $\mathbf{R}_j^{attn}$
$\tilde{F}_{o_\theta}^{s}(\mathbf{R}_j^{attn})$	:	the salience component in terms of orientation in θ of a local post-attentive feature of $\mathbf{R}_j^{attn}$
$\tilde{F}_{o_{0^\circ}}^{s}(\mathbf{R}_j^{attn})$	:	the salience component in terms of orientation in 0° of a local post-attentive feature of $\mathbf{R}_j^{attn}$
$\tilde{F}_{o_{45^\circ}}^{s}(\mathbf{R}_j^{attn})$	:	the salience component in terms of orientation in 45° of a local post-attentive feature of $\mathbf{R}_j^{attn}$
$\tilde{F}_{o_{90^\circ}}^{s}(\mathbf{R}_j^{attn})$	:	the salience component in terms of orientation in 90° of a local post-attentive feature of $\mathbf{R}_j^{attn}$
$\tilde{F}_{o_{135^\circ}}^{s}(\mathbf{R}_j^{attn})$	:	the salience component in terms of orientation in 135° of a local post-attentive feature of $\mathbf{R}_j^{attn}$
$\tilde{F}_{rg}^{s}(\mathbf{R}_j^{attn})$	:	the salience component in terms of red-green pair of a local post-attentive feature of $\mathbf{R}_j^{attn}$
$\mathbf{F}$	:	a pre-attentive feature (vector)
$\mathbf{F}(l)$	:	a pre-attentive feature (vector) at level l
$\mathbf{F}_{by}$	:	the pre-attentive feature in terms of blue-yellow pair
$\mathbf{F}_{ct}$	:	the pre-attentive feature in terms of contour
$\mathbf{F}_{int}$	:	the pre-attentive feature in terms of intensity
$\mathbf{F}_{mv}$	:	the pre-attentive feature in terms of motion energy
$\mathbf{F}_{o_\theta}$	:	the pre-attentive feature in terms of orientation energy in orientation θ

$\mathbf{F}_{rg}$	:	the pre-attentive feature in terms of red-green pair
$\hat{\mathbf{F}}_i^0$	:	the aggregate feature (vector) of a vertex v_i at level $l = 0$ in an irregular pyramid
$\hat{\mathbf{F}}_j^0$	:	the aggregate feature (vector) of a vertex v_j at level $l = 0$ in an irregular pyramid
$\hat{\mathbf{F}}_i^l$	:	the aggregate feature (vector) of a vertex v_i at level l in an irregular pyramid
$\hat{\mathbf{F}}_j^l$	:	the aggregate feature (vector) of a vertex v_j at level l in an irregular pyramid
$\mathbf{F}'(l_c, l_s)$	:	a center-surround difference map (vector)
$\mathbf{F}'_{by}(l_c, l_s)$	:	the center-surround difference map (vector) in terms of blue-yellow pair
$\mathbf{F}'_{int}(l_c, l_s)$	:	the center-surround difference map (vector) in terms of intensity
$\mathbf{F}'_{mv}(l_c, l_s)$	:	the center-surround difference map (vector) in terms of motion energy
$\mathbf{F}'_{o_\theta}(l_c, l_s)$	:	the center-surround difference map (vector) in terms of orientation energy in preferred direction θ
$\mathbf{F}'_{rg}(l_c, l_s)$	:	the center-surround difference map (vector) in terms of red-green pair
$\mathbf{F}^s$	:	a conspicuity map (vector)
$\mathbf{F}^s_{by}$	:	the conspicuity map (vector) in terms of blue-yellow pair
$\mathbf{F}^s_{ct}$	:	the conspicuity map (vector) in terms of contour
$\mathbf{F}^s_{int}$	:	the conspicuity map (vector) in terms of intensity
$\mathbf{F}^s_{mv}$	:	the conspicuity map (vector) in terms of motion energy
$\mathbf{F}^s_{o_\theta}$	:	the conspicuity map (vector) in terms of orientation energy in a preferred direction θ
$\mathbf{F}^s_{rg}$	:	the conspicuity map (vector) in terms of red-green pair
$\mathbf{F}^t$	:	an attentional template (vector)

$\{\mathbf{F}^t\}$	:	the set of attentional templates in terms of all task-relevant feature dimensions
$\mathbf{F}^t_{by}$	:	the attentional template in terms of blue-yellow pair
$\mathbf{F}^t_{ct}$	:	the attentional template in terms of contour
$\mathbf{F}^{t,\mu}_{ct}$	:	means of positions of control points in the attentional template in terms of contour
$\mathbf{F}^{t,\mu,x}_{ct}$	:	means of x-coordinates of control points in the attentional template in terms of contour
$\mathbf{F}^{t,\mu,y}_{ct}$	:	means of y-coordinates of control points in the attentional template in terms of contour
$\mathbf{F}^{t,\sigma}_{ct}$	:	STDs of positions of control points in the attentional template in terms of contour
$\mathbf{F}^{t,\sigma,x}_{ct}$	:	STDs of x-coordinates of control points in the attentional template in terms of contour
$\mathbf{F}^{t,\sigma,y}_{ct}$	:	STDs of y-coordinates of control points in the attentional template in terms of contour
$\mathbf{F}^t_{int}$	:	the attentional template in terms of intensity
$\mathbf{F}^t_{rg}$	:	the attentional template in terms of red-green pair
$\tilde{\mathbf{F}}$	:	a post-attentive feature (vector)
$\tilde{\mathbf{F}}^a$	:	an appearance component of a post-attentive feature (vector)
$\tilde{\mathbf{F}}^s$	:	a salience component of a post-attentive feature (vector)
$\tilde{\mathbf{F}}_{gb}$	:	a global post-attentive feature (vector)
$\{\tilde{\mathbf{F}}_{gb}\}$	:	the set of global post-attentive features at all global control points
$\tilde{\mathbf{F}}_{gb}(\mathbf{r}_{cp})$	:	a global post-attentive feature at a global control point $\mathbf{r}_{cp}$
$\tilde{\mathbf{F}}^a_{gb}(\mathbf{r}_{cp})$	:	the appearance component of a global post-attentive feature at a global control point $\mathbf{r}_{cp}$
$\tilde{\mathbf{F}}^s_{gb}(\mathbf{r}_{cp})$	:	the salience component of a global post-attentive feature at a global control point $\mathbf{r}_{cp}$

$\tilde{\mathbf{F}}_{lc}$	:	a local post-attentive feature (vector)
$\tilde{\mathbf{F}}_{lc}^{ld}$	:	the vector that is identical to $\tilde{\mathbf{F}}_{lc}$, except that appearance histogram measures are removed in $\tilde{\mathbf{F}}_{lc}^{ld}$
$\{\tilde{\mathbf{F}}_{lc}\}$	:	the set of local post-attentive features of all proto-objects in the complete region being attended
$\tilde{\mathbf{F}}_{lc}(\mathbf{R}_j^{attn})$	:	a local post-attentive feature (vector) in a proto-object $\mathbf{R}_j^{attn}$ in the complete region being attended
$\tilde{\mathbf{F}}_{lc}^{a}(\mathbf{R}_j^{attn})$	:	the appearance component of a local post-attentive feature (vector) of $\mathbf{R}_j^{attn}$
$\tilde{\mathbf{F}}_{lc}^{s}(\mathbf{R}_j^{attn})$	:	the salience component of a local post-attentive feature (vector) of $\mathbf{R}_j^{attn}$
$\tilde{\mathbf{F}}_{by}^{a}(\mathbf{R}_j^{attn})$	:	the appearance component in terms of blue-yellow pair of a local post-attentive feature (vector) of $\mathbf{R}_j^{attn}$
$\tilde{\mathbf{F}}_{int}^{a}(\mathbf{R}_j^{attn})$	:	the appearance component in terms of intensity of a local post-attentive feature (vector) of $\mathbf{R}_j^{attn}$
$\tilde{\mathbf{F}}_{o}^{a}(\mathbf{R}_j^{attn})$	:	the appearance component in terms of local orientations of a local post-attentive feature (vector) of $\mathbf{R}_j^{attn}$
$\tilde{\mathbf{F}}_{rg}^{a}(\mathbf{R}_j^{attn})$	:	the appearance component in terms of red-green pair of a local post-attentive feature (vector) of $\mathbf{R}_j^{attn}$
g	:	the index of proto-objects
$g_\theta(x, y)$	:	the 2-D Gabor filter function in orientation θ
$g_{\theta,0}(x, y)$	:	the even-symmetric part of a 2-D Gabor filter function in orientation θ
$g_{\theta,-\frac{1}{2}\pi}(x, y)$	:	the odd-symmetric part of a 2-D Gabor filter function in orientation θ
$G_\theta(u, v)$	:	Fourier Transform of a 2-D Gabor filter function in orientation θ
$G_{\theta,0}(u, v)$	:	Fourier transform of the even-symmetric part of a 2-D Gabor filter function in orientation θ

$G_{\theta,-\frac{1}{2}\pi}(u,v)$	:	Fourier transform of the odd-symmetric part of a 2-D Gabor filter function in orientation θ
$\mathbf{g}$	:	green channel in the Red-Green-Blue color model
$\mathbf{G}$	:	green channel of the broadly-tuned color channels
$\mathbf{G}_l$	:	the graph used to represent the level l in an irregular pyramid
$\mathbf{G}_{l+1}$	:	the graph used to represent the level $l+1$ in an irregular pyramid
$\mathbf{H}_{by}^{a,j}$	:	the appearance histogram in terms of blue-yellow pair of $\mathbf{R}_j^{attn}$
$\mathbf{H}_{int}^{a,j}$	:	the appearance histogram in terms of intensity of $\mathbf{R}_j^{attn}$
$\mathbf{H}_{o}^{a,j}$	:	the appearance histogram in terms of local orientations of $\mathbf{R}_j^{attn}$
$\mathbf{H}_{rg}^{a,j}$	:	the appearance histogram in terms of red-green pair of $\mathbf{R}_j^{attn}$
i_{max}	:	the index of the instance or the control point that has the maximal posterior probability
$\mathbf{I}$	:	an image
j_{max}	:	the index of the part or the contour instance that has the maximal posterior probability
k_{max}	:	the index of the LTM object that has the maximal posterior probability
K	:	the magnitude of the Gaussian envelope of a 2-D Gabor filter
$\mathbf{K}_\omega$	:	a 6×6 diagonal coefficient matrix
l	:	a level in a Gaussian pyramid (i.e., spatial scale) or a level in an irregular pyramid
l_c	:	a center scale used in the bottom-up competition module
l_s	:	a surround scale used in the bottom-up competition module
l_{top}	:	the level at which each vertex has no neighbors in an irregular pyramid
l_{wk}	:	the working scale
m_v	:	an innovation on a measurement line of a predicted contour curve
m_z	:	an observation on a measurement line of a predicted contour curve

$\overline{m}$	:	the average of all local maxima in a center-surround difference map
M_{mn}	:	moments of a proto-object
$\overline{M}_{11}, \overline{M}_{20}, \overline{M}_{02}$	:	2-order moments relative to the center of mass of a proto-object
n	:	the number of vertices at a level in an irregular pyramid used in the pre-attentive segmentation
n_{rel}	:	the index of the task-relevant part or the index of the task-relevant contour instance
nN	:	the number of negative objects in the testing image set
nP	:	the number of positive objects in the testing image set
$nTOT$	:	the total number of frames in a tracking video
nTP	:	the number of frames in which the target is correctly detected
N_c	:	the number of predicted contour curves used to estimate the top-down bias in terms of contour
N_{cfp}	:	the number of contour feature points along a measurement line of a predicted contour curve
N_{cp}^{n}	:	the number of control points along the contour instance indexed by n
$N_{cp}^{n_{rel}}$	:	the number of control points of a contour instance indexed by n_{rel}
N_{ct}	:	the number of contour instances of a global coding of an LTM object representation
N_d	:	the number of iterations of the decimation in the pre-attentive segmentation
N_f	:	the number of features used in the pre-attentive segmentation
N_g	:	the number of pixels of a proto-object indexed by g
$N_{L1}^{gb}(j,k)$	:	the number of control points belonging to the contour instance j in the global PNN of the object k
$N_{L1,max}^{gb}$	:	the maximal number of control points in a global PNN

$N_{L1}^{lc}(j,k)$	:	the number of instances belonging to the part j in the local PNN of the object k
$N_{L1,max}^{lc}$	:	the maximal number of instances in a local PNN
$N_{L2}^{gb}(k)$	:	the number of contour instances in the global PNN of the object k
$N_{L2,max}^{gb}$	:	the maximal number of contour instances in a global PNN
$N_{L2}^{lc}(k)$	:	the number of parts in the local PNN of the object k
$N_{L2,max}^{lc}$	:	the maximal number of parts in a local PNN
N_{L3}	:	the number of existing LTM objects
N_{ml}	:	the number of measurement lines along one predicted curve
N_{nb}	:	the number of neighbors of a vertex in an irregular pyramid used in the pre-attentive segmentation
N_p	:	the number of parts of an object
N_{ptn}	:	the entry number of the set $\{\tilde{\mathbf{F}}_{lc}\}$, the set $\{\tilde{\mathbf{F}}_{gb}\}$ or the total entry number of $\{\tilde{\mathbf{F}}_{lc}\}$ and $\{\tilde{\mathbf{F}}_{gb}\}$
$\mathcal{N}(\cdot)$	:	a normalization operator
$\mathbb{N}_i$	:	the neighbor set of a vertex v_i in a graph
$\mathbb{N}_j$	:	the neighbor set of a vertex v_j in a graph
$\mathbb{N}_{\mathbf{r}_i}$	:	neighbors of a pixel $\mathbf{r}_i$
$\mathbf{O}$	:	an LTM object representation
$\mathbf{O}_{attn}$	:	the matched LTM object representation given the attended proto-object
$\mathbf{O}_{by}$	:	the local coding of an LTM object representation in terms of blue-yellow pair
$\mathbf{O}_{ct}$	:	the global coding of an LTM object representation in terms of contour
$\mathbf{O}_{gb}$	:	the global coding of an LTM object representation
$\mathbf{O}_{int}$	:	the local coding of an LTM object representation in terms of intensity

$\mathbf{O}_{lc}$	:	the local coding of an LTM object representation
$\mathbf{O}_{o_{0°}}$	:	the local coding of an LTM object representation in terms of local orientation in 0°
$\mathbf{O}_{o_{45°}}$	:	the local coding of an LTM object representation in terms of local orientation in 45°
$\mathbf{O}_{o_{90°}}$	:	the local coding of an LTM object representation in terms of local orientation in 90°
$\mathbf{O}_{o_{135°}}$	:	the local coding of an LTM object representation in terms of local orientation in 135°
$\mathbf{O}_{rg}$	:	the local coding of an LTM object representation in terms of red-green pair
$\mathbf{O}^{a}$	:	an appearance descriptor of an LTM object representation
$\mathbf{O}^{a}_{by}$	:	the local appearance descriptor in terms of blue-yellow pair of an LTM object representation
$\mathbf{O}^{a}_{ct}$	:	the appearance descriptor in a global coding of an LTM object representation
$\mathbf{O}^{a,1}_{ct}$	:	the contour instance (indexed by 1) in $\mathbf{O}^{a}_{ct}$
$\mathbf{O}^{a,2}_{ct}$	:	the contour instance (indexed by 2) in $\mathbf{O}^{a}_{ct}$
$\mathbf{O}^{a,n}_{ct}$	:	the contour instance (indexed by n and $n = 1, 2, ..., N_{ct}$) in $\mathbf{O}^{a}_{ct}$
$\mathbf{O}^{a,N_{ct}}_{ct}$	:	the contour instance (indexed by N_{ct}) in $\mathbf{O}^{a}_{ct}$
$\mathbf{O}^{a}_{int}$	:	the local appearance descriptor in terms of intensity of an LTM object representation
$\mathbf{O}^{a}_{o_\theta}$	:	the local appearance descriptor in terms of local orientation θ of an LTM object representation
$\mathbf{O}^{a}_{rg}$	:	the local appearance descriptor in terms of red-green pair of an LTM object representation
$\mathbf{O}^{s}$	:	a salience descriptor of an LTM object representation

$\mathbf{O}^{s}_{ct}$	:	the salience descriptor in a global coding of an LTM object representation
$\mathbf{O}^{s}_{int}$	:	the salience descriptor in terms of intensity in a local coding of an LTM object representation
$\mathbf{O}^{L1}_{gb,k}$	:	the control point level of the global coding of the LTM object representation indexed by k
$\mathbf{O}^{L1,a}_{gb,k}$	:	the appearance component of $\mathbf{O}^{L1}_{gb,k}$
$\mathbf{O}^{L2}_{gb,k}$	:	the contour instance level of the global coding of the LTM object representation indexed by k
$\mathbf{O}^{L3}_{gb,k}$	:	the object level of the global coding of the LTM object representation indexed by k
$\mathbf{O}^{ld}_{gb,k}$	:	the low-level global coding of an LTM object representation indexed by k
$\mathbf{O}^{ld,a}_{gb,k}$	:	the appearance component of $\mathbf{O}^{ld}_{gb,k}$
$\mathbf{O}^{ld,s}_{gb,k}$	:	the salience component of $\mathbf{O}^{ld}_{gb,k}$
$\mathbf{O}^{L1}_{lc,k}$	:	the instance level of the local coding of the LTM object representation indexed by k
$\mathbf{O}^{L2}_{lc,k}$	:	the part level of the local coding of the LTM object representation indexed by k
$\mathbf{O}^{L3}_{lc,k}$	:	the object level of the local coding of the LTM object representation indexed by k
$\mathbf{O}^{ld}_{lc}$	:	the low-level local coding of an LTM object representation indexed by k
$\mathcal{O}$	:	computational complexity
$p_{attn}(\mathbf{r}_i)$	:	the probability of the location $\mathbf{r}_i$ being attended
$p_{attn}(\mathbf{R}_g)$	:	the probability of a proto-object (indexed by g) being attended
$p_{bu}(\mathbf{r}_i)$	:	the probability of a spatial location $\mathbf{r}_i$ being attended by the bottom-up attention mechanism

$p^{[l,l+1]}$: an inter-level edge between levels l and $l+1$ in an irregular pyramid

$p_{i,i}^{[l,l+1]}$: the inter-level edge between a vertex v_i at level l and a vertex v_i at level $l+1$ in an irregular pyramid, i.e., the vertex v_i survives at level $l+1$

$p_{i,k}^{[l,l+1]}$: the inter-level edge between a vertex v_i at level l and a vertex v_k at level $l+1$ in an irregular pyramid

$p_{j,k}^{[l,l+1]}$: the inter-level edge between a vertex v_j at level l and a vertex v_k at level $l+1$ in an irregular pyramid

$p_k(\tilde{\mathbf{F}}_{gb})$: the probabilistic mixture estimation of the object k in its global PNN

$p_k(\tilde{\mathbf{F}}_{lc})$: the probabilistic mixture estimation of the object k in its local PNN

$\overline{p}_k(\tilde{\mathbf{F}}_{gb})$: the probabilistic summary estimation of the object k in its global PNN

$\overline{p}_k(\tilde{\mathbf{F}}_{lc})$: the probabilistic summary estimation of the object k in its local PNN

$\overline{\overline{p}}_k(\{\tilde{\mathbf{F}}_{gb}\})$: the recognition probability of the united training pattern $\{\tilde{\mathbf{F}}_{gb}\}$ at the object level of a global PNN

$\overline{\overline{p}}_k(\tilde{\mathbf{F}}_{lc})$: the recognition probability of the training pattern $\tilde{\mathbf{F}}_{lc}$ at the object level of a local PNN

$\tilde{p}(\mathbf{X})$: the truncated part of a Gaussian distribution given the input vector $\mathbf{X}$

$\tilde{p}_{k_{max}}(\tilde{\mathbf{F}}_{gb})$: the truncated part of the probabilistic mixture estimation $p_{k_{max}}(\tilde{\mathbf{F}}_{gb})$

$\tilde{p}_{k_{max}}(\tilde{\mathbf{F}}_{lc})$: the truncated part of the probabilistic mixture estimation $p_{k_{max}}(\tilde{\mathbf{F}}_{lc})$

$p_{td}(\mathbf{r}_i)$: the prior probability of a spatial location $\mathbf{r}_i$ being attended by the top-down attention mechanism

$p_{td}(\mathbf{r}_i|\mathbf{F}^t)$: the posterior probability of a spatial location $\mathbf{r}_i$ being attended by the top-down attention mechanism

$p_{td}(\mathbf{r}_i|\{\mathbf{F}^t\})$: the posterior probability of a location $\mathbf{r}_i$ being attended by top-down attention in terms of all task-relevant feature dimensions

$p_{td}(\mathbf{r}_i|\mathbf{F}^t_{by})$: the posterior of a spatial location $\mathbf{r}_i$ being attended by the top-down attention mechanism in the case that blue-yellow pair is the task-relevant feature

$p_{td}(\mathbf{r}_i|\mathbf{F}^t_{ct})$: the posterior of a spatial location $\mathbf{r}_i$ being attended by the top-down attention mechanism in the case that contour is the task-relevant feature

$p_{td}(\mathbf{r}_i|\mathbf{F}^t_{int})$: the posterior of a spatial location $\mathbf{r}_i$ being attended by the top-down attention mechanism in the case that intensity is the task-relevant feature

$p_{td}(\mathbf{r}_i|\mathbf{F}^t_{mv})$: the posterior of a spatial location $\mathbf{r}_i$ being attended by the top-down attention mechanism in the case that motion is the task-relevant feature

$p_{td}(\mathbf{r}_i|\mathbf{F}^t_{o_\theta})$: the posterior of a spatial location $\mathbf{r}_i$ being attended by the top-down attention mechanism in the case that orientation in θ is the task-relevant feature

$p_{td}(\mathbf{r}_i|\mathbf{F}^t_{rg})$: the posterior of a spatial location $\mathbf{r}_i$ being attended by the top-down attention mechanism in the case that red-green pair is the task-relevant feature

$p_{td}(\mathbf{C}^m_{\mathbf{R}_g})$: the prior probability of a predicted curve $\mathbf{C}^m_{\mathbf{R}_g}$

$p_{td}(\mathbf{C}^m_{\mathbf{R}_g}|\mathbf{R}_g)$: the posterior probability of each predicted contour $\mathbf{C}^m_{\mathbf{R}_g}$

$p_{td}(\mathbf{F}^t|\mathbf{r}_i)$: the observation likelihood of a spatial location $\mathbf{r}_i$ being attended by the top-down attention mechanism

$p_{td}(\{\mathbf{F}^t\}|\mathbf{r}_i)$: the observation likelihood of a location $\mathbf{r}_i$ being attended by top-down attention in terms of all task-relevant feature dimensions

$p_{td}(\mathbf{F}_{by}^t|\mathbf{r}_i)$: the observation likelihood of a spatial location $\mathbf{r}_i$ being attended by the top-down attention mechanism in the case that blue-yellow pair is the task-relevant feature

$p_{td}(\mathbf{F}_{ct}^t|\mathbf{r}_i)$: the observation likelihood of a spatial location $\mathbf{r}_i$ being attended by the top-down attention mechanism in the case that contour is the task-relevant feature

$p_{td}(\mathbf{F}_{ct}^t|\mathbf{R}_g)$: the observation likelihood of a proto-object $\mathbf{R}_g$ being attended by the top-down attention mechanism in the case that contour is the task-relevant feature

$p_{td}(\mathbf{F}_{int}^t|\mathbf{r}_i)$: the observation likelihood of a spatial location $\mathbf{r}_i$ being attended by the top-down attention mechanism in the case that intensity is the task-relevant feature

$p_{td}(\mathbf{F}_{mv}^t|\mathbf{r}_i)$: the observation likelihood of a spatial location $\mathbf{r}_i$ being attended by the top-down attention mechanism in the case that motion is the task-relevant feature

$p_{td}(\mathbf{F}_{o\theta}^t|\mathbf{r}_i)$: the observation likelihood of a spatial location $\mathbf{r}_i$ being attended by the top-down attention mechanism in the case that orientation in θ is the task-relevant feature

$p_{td}(\mathbf{F}_{rg}^t|\mathbf{r}_i)$: the observation likelihood of a spatial location $\mathbf{r}_i$ being attended by the top-down attention mechanism in the case that red-green pair is the task-relevant feature

$p_{td}(\mathbf{R}_g|\mathbf{C}_{\mathbf{R}_g}^m)$: the observation likelihood of a predicted contour curve $\mathbf{C}_{\mathbf{R}_g}^m$

$p_{td}(\mathbf{R}_g|m_v, \mathbf{C}_{\mathbf{R}_g}^m)$: the observation likelihood of a single measurement line of a predicted contour curve

$p(\boldsymbol{\psi}|\mathbf{d}_s, d_t)$: the conditional distribution of the optical flow given the spatial and temporal derivatives

P_{TPR} : tracking precision

$q_i^{j,k}(\tilde{\mathbf{F}}_{gb})$: the probability density of the control point i along the contour instance j in the global PNN of the object k

$q_i^{j,k}(\tilde{\mathbf{F}}_{lc})$: the probability density of the instance i of the part j in the local PNN of the object k

$\overline{\overline{q}}_i^{j,k}(\tilde{\mathbf{F}}_{gb})$: the recognition probability of a single training pattern $\tilde{\mathbf{F}}_{gb}$ at the control point level of a global PNN

$\overline{\overline{q}}_i^{j,k}(\tilde{\mathbf{F}}_{lc})$: the recognition probabilities of the training pattern $\tilde{\mathbf{F}}_{lc}$ at the instance level of a local PNN

$\tilde{q}_{i_{max}}^{j,k}(\tilde{\mathbf{F}}_{gb})$: the truncated part of the probability $q_{i_{max}}^{j,k}(\tilde{\mathbf{F}}_{gb})$

$\tilde{q}_{i_{max}}^{j,k}(\tilde{\mathbf{F}}_{lc})$: the truncated part of the probability $q_{i_{max}}^{j,k}(\tilde{\mathbf{F}}_{lc})$

$\mathbf{Q}_0$: the control point vector of a contour

$r_j^k(\tilde{\mathbf{F}}_{gb})$: the probabilistic mixture estimation of the contour instance j in the global PNN of the object k

$r_j^k(\tilde{\mathbf{F}}_{lc})$: the probabilistic mixture estimation of the part j in the local PNN of the object k

$\overline{r}_j^k(\tilde{\mathbf{F}}_{gb})$: the probabilistic summary estimation of the contour instance j in the global PNN of the object k and it is a multi-dimensional Gaussian distribution

$\overline{r}_j^k(\tilde{\mathbf{F}}_{lc})$: the probabilistic summary estimation of the part j in the local PNN of the object k and it is a multi-dimensional Gaussian distribution

$\overline{\overline{r}}_j^k(\{\tilde{\mathbf{F}}_{gb}\})$: the recognition probability of the united training pattern $\{\tilde{\mathbf{F}}_{gb}\}$ at the contour instance level of a global PNN

$\overline{\overline{r}}_j^k(\tilde{\mathbf{F}}_{lc})$: the recognition probability of the training pattern $\tilde{\mathbf{F}}_{lc}$ at the part level of a local PNN

$\tilde{r}_{j_{max}}^k(\tilde{\mathbf{F}}_{gb})$: the truncated part of the probabilistic mixture estimation $r_{j_{max}}^k(\tilde{\mathbf{F}}_{gb})$

$\tilde{r}_{j_{max}}^k(\tilde{\mathbf{F}}_{lc})$: the truncated part of the probabilistic mixture estimation $r_{j_{max}}^k(\tilde{\mathbf{F}}_{lc})$

$r_{\theta,0}(\mathbf{r}_i, l)$	:	the convolution result of the even-symmetric part of a 2-D Gabor filter in orientation θ at pixel $\mathbf{r}_i$ at scale l
$r_{\theta,-\frac{1}{2}\pi}$	:	the convolution result of the odd-symmetric part of a 2-D Gabor filter in orientation θ at pixel $\mathbf{r}_i$ at scale l
$\mathbf{r}$	:	red channel in the Red-Green-Blue color model
$\{\mathbf{r}_a^j\}$	:	the set of pixels with available orientation in a $\mathbf{R}_j^{attn}$
$\{\mathbf{r}_{cp}\}$	:	the set of control points of the attended object
$\mathbf{r}_i$	:	the pixel indexed by i in an image
$\mathbf{R}$	:	red channel of the broadly-tuned color channels
$\{\mathbf{R}_{attn}\}$	:	the complete region being attended, i.e., a set of all proto-objects in the complete region
$\mathbf{R}_c$	:	a band range that is set manually along the object's global contour in the reference frame to filter out local control points
$\mathbf{R}_g$	:	a proto-object indexed by g
$\{\mathbf{R}_g\}$	:	the set of proto-objects
$\mathbf{R}_j$	:	the attended proto-object
$\mathbf{R}_j^{attn}$	:	a proto-object in the complete region being attended
$\{\mathbf{R}_k\}$	:	the set of neighbor proto-objects around the attended proto-object $\mathbf{R}_j$
$S_{bu}(\mathbf{r}_i)$	:	the location-based bottom-up saliency (scalar) at the location $\mathbf{r}_i$
$\mathbf{S}_{bu}$	:	a location-based bottom-up saliency map (vector)
TP	:	the number of true positives
(u_0, v_0)	:	the spatial center frequencies of a 2-D Gabor filter
$[u_{i,1}^l, ..., u_{i,g}^l, ...]$	:	the membership vector of a vertex v_i at level l and each entry of this vector represents the membership of v_i to a proto-object $\mathbf{R}_g$
$u_{i,g}^{l-1}$	:	an entry of the membership vector $[u_{i,1}^{l-1}, ..., u_{i,g}^{l-1}, ...]$ of a vertex v_i at level $l-1$

$u^{l}_{k,g}$	:	an entry of the membership vector $[u^{l}_{k,1}, ..., u^{l}_{k,g}, ...]$ of a vertex v_k at level l
$u^{l_{top}}_{k,g}$	:	an entry of the membership vector $[u^{l_{top}}_{k,1}, ..., u^{l_{top}}_{k,g}, ...]$ of a vertex v_k at level l_{top}
v	:	a vertex in a graph
v_i	:	the vertex indexed by i at a level in an irregular pyramid
v_j	:	the vertex indexed by j at a level in an irregular pyramid
$v_{j'}$	:	the vertex indexed by j' at a level in an irregular pyramid
v_k	:	the vertex indexed by k at a level in an irregular pyramid
$v_{k'}$	:	the vertex indexed by k' at a level in an irregular pyramid
$\mathbf{V}_l$	:	the vertex set in the graph $\mathbf{G}_l$
$\mathbf{V}_{l+1}$	:	the vertex set in the graph $\mathbf{G}_{l+1}$
$w(m)$	:	the 1-D weighting function used to generate a 1-D Gaussian pyramid
$w(m,n)$	:	the 2-D weighting function used to generate a 2-D Gaussian pyramid
w_{bu}	:	the weight of bottom-up attention
w_{td}	:	the weight of top-down attention
$\mathbf{W}$	:	the shape factor of a contour
(x,y)	:	the spatial coordinates of a pixel in an image
(x_0,y_0)	:	the centroid of the Gaussian envelope of a 2-D Gabor filter
x_1,x_2,x_3,x_4,x_5,x_6	:	entries of the shape state vector (i.e., $\mathbf{X}$) of a contour
x'_1,x'_2,x'_3,x'_4,x'_5,x'_6	:	entries of $\mathbf{X}'_{\mathbf{R}_g}$
$\mathbf{X}$	:	the shape state vector of a contour
$\mathbf{X}^{m}_{\mathbf{R}_g}$	:	the shape state vector of a predicted contour (indexed by m) for a proto-object (indexed by g) used to estimate the top-down bias in terms of contour

$\mathbf{X}'_{\mathbf{R}_g}$	:	the deterministic prediction of the shape state vector for a proto-object (indexed by g) used to estimate the top-down bias in terms of contour
$\mathbf{Y}$	:	yellow channel of the broadly-tuned color channels
$\aleph(\cdot)$	:	a normal distribution
δ	:	the difference between a center scale and a surround scale
$\mu_i^{j,k}$	:	an entry of the mean vector $\boldsymbol{\mu}_i^{j,k}$
$\mu_{i,d}^{j,k}$	:	the value in terms of the feature dimension d in $\mu_i^{j,k}$
$\mu_{by}^{a,1}$	:	the mean of a part (indexed by 1) in $\mathbf{O}_{by}^a$
$\mu_{by}^{a,2}$	:	the mean of a part (indexed by 2) in $\mathbf{O}_{by}^a$
μ_{by}^{a,N_p}	:	the mean of a part (indexed by N_p) in $\mathbf{O}_{by}^a$
$\mu_{int}^{a,1}$	:	the mean of a part (indexed by 1) in $\mathbf{O}_{int}^a$
$\mu_{int}^{a,2}$	:	the mean of a part (indexed by 2) in $\mathbf{O}_{int}^a$
μ_{int}^{a,N_p}	:	the mean of a part (indexed by N_p) in $\mathbf{O}_{int}^a$
$\mu_{o_\theta}^{a,1}$	:	the mean of a part (indexed by 1) in $\mathbf{O}_{o_\theta}^a$
$\mu_{o_\theta}^{a,2}$	:	the mean of a part (indexed by 2) in $\mathbf{O}_{o_\theta}^a$
$\mu_{o_\theta}^{a,N_p}$	:	the mean of a part (indexed by N_p) in $\mathbf{O}_{o_\theta}^a$
$\mu_{rg}^{a,1}$	:	the mean of a part (indexed by 1) in $\mathbf{O}_{rg}^a$
$\mu_{rg}^{a,2}$	:	the mean of a part (indexed by 2) in $\mathbf{O}_{rg}^a$
μ_{rg}^{a,N_p}	:	the mean of a part (indexed by N_p) in $\mathbf{O}_{rg}^a$
$\mu_{x,n}^{a,1}, \mu_{y,n}^{a,1}$	:	the mean of the spatial position of a control point (indexed by 1) along a contour instance (indexed by n)
$\mu_{x,n}^{a,N_{cp}^n}, \mu_{y,n}^{a,N_{cp}^n}$	:	the mean of the spatial position of a control point (indexed by N_{cp}^n) along a contour instance (indexed by n)
$\mu_{by}^{s,1}$	:	the mean of conspicuity values in terms of blue-yellow of a part (indexed by 1) in $\mathbf{O}_{by}^s$
$\mu_{by}^{s,2}$	:	the mean of conspicuity values in terms of blue-yellow of a part (indexed by 2) in $\mathbf{O}_{by}^s$

μ_{by}^{s,N_p} : the mean of conspicuity values in terms of blue-yellow of a part (indexed by N_p) in $\mathbf{O}_{by}^{s}$

$\mu_{ct}^{s,1}$: the mean of the conspicuity values in terms of the contour feature over all control points along a contour instance (indexed by 1)

$\mu_{ct}^{s,2}$: the mean of the conspicuity values in terms of the contour feature over all control points along a contour instance (indexed by 2)

$\mu_{ct}^{s,N_{ct}}$: the mean of the conspicuity values in terms of the contour feature over all control points along a contour instance (indexed by N_{ct})

$\mu_{f}^{s,n}$: a salience descriptor in terms of a feature dimension f of a part indexed by n or of a contour instance indexed by n

$\mu_{int}^{s,1}$: the mean of conspicuity values in terms of intensity of a part (indexed by 1) in $\mathbf{O}_{int}^{s}$

$\mu_{int}^{s,2}$: the mean of conspicuity values in terms of intensity of a part (indexed by 2) in $\mathbf{O}_{int}^{s}$

μ_{int}^{s,N_p} : the mean of conspicuity values in terms of intensity of a part (indexed by N_p) in $\mathbf{O}_{int}^{s}$

$\mu_{o_\theta}^{s,1}$: the mean of conspicuity values in terms of orientation in θ of a part (indexed by 1) in $\mathbf{O}_{o_\theta}^{s}$

$\mu_{o_\theta}^{s,2}$: the mean of conspicuity values in terms of orientation in θ of a part (indexed by 2) in $\mathbf{O}_{o_\theta}^{s}$

$\mu_{o_\theta}^{s,N_p}$: the mean of conspicuity values in terms of orientation in θ of a part (indexed by N_p) in $\mathbf{O}_{o_\theta}^{s}$

$\mu_{rg}^{s,1}$: the mean of conspicuity values in terms of red-green of a part (indexed by 1) in $\mathbf{O}_{rg}^{s}$

$\mu_{rg}^{s,2}$: the mean of conspicuity values in terms of red-green of a part (indexed by 2) in $\mathbf{O}_{rg}^{s}$

μ_{rg}^{s,N_p} : the mean of conspicuity values in terms of red-green of a part (indexed by N_p) in $\mathbf{O}_{rg}^{s}$

$\tilde{\mu}_{by}^{a,j}$	:	the appearance mean in terms of blue-yellow pair of $\mathbf{R}_j^{attn}$
$\tilde{\mu}_{int}^{a,j}$	:	the appearance mean in terms of intensity of $\mathbf{R}_j^{attn}$
$\tilde{\mu}_{rg}^{a,j}$	:	the appearance mean in terms of red-green pair of $\mathbf{R}_j^{attn}$
$\tilde{\mu}_{by}^{s,j}$	:	the salience mean in terms of blue-yellow pair of $\mathbf{R}_j^{attn}$
$\tilde{\mu}_{int}^{s,j}$	:	the salience mean in terms of intensity of $\mathbf{R}_j^{attn}$
$\tilde{\mu}_{o\theta}^{s,j}$	:	the salience mean in terms of orientation in θ of $\mathbf{R}_j^{attn}$
$\tilde{\mu}_{rg}^{s,j}$	:	the salience mean in terms of red-green pair of $\mathbf{R}_j^{attn}$
$\overline{\mu}$	:	the mean of a probabilistic summary of a set of Gaussian distributions
$\overline{\mu}_j^k$	:	an entry of the mean vector $\overline{\boldsymbol{\mu}}_j^k$
$\boldsymbol{\mu}_1$	:	the mean vector of one Gaussian distribution used to calculate Bhattacharyya distance
$\boldsymbol{\mu}_2$	:	the mean vector of the other Gaussian distribution used to calculate Bhattacharyya distance
$\boldsymbol{\mu}_i^{j,k}$	:	the mean vector of the instance i of the part j in the local PNN of the object k or the mean vector of the control point i along the contour instance j in the global PNN of the object k
$\boldsymbol{\mu}_{\psi}$	:	the mean vector of the conditional distribution of the optical flow
$\overline{\boldsymbol{\mu}}_j^k$	:	the mean vector of $\overline{r}_j^k(\tilde{\mathbf{F}}_{lc})$ or the mean vector of $\overline{r}_j^k(\tilde{\mathbf{F}}_{gb})$
$\ominus$	:	across-scale subtraction
$\bigoplus$	:	across-scale addition
$\boldsymbol{\omega}^m$	:	a 6×1 random vector whose entries are normally distributed (the mean is 0 and the STD is 1 for each entry)
ϕ_a	:	the area threshold used in the pre-attentive segmentation algorithm
ϕ_e	:	the similarity threshold used in the similarity-driven neighbor search procedure

$\pi_i^{j,k}$: the occurrence rate of the instance i of the part j in the local PNN of the object k or the occurrence rate of the control point i along the contour instance j in the global PNN of the object k

$\boldsymbol{\psi}$: the optical flow vector

φ_j^k : the contribution of a part to the object k or the contribution of a contour instance to the object k

$\sigma_i^{j,k}$: an entry of the STD vector $\boldsymbol{\sigma}_i^{j,k}$

$\sigma_{i,d}^{j,k}$: the STD in terms of a feature dimension d in $\sigma_i^{j,k}$

$\sigma_{by}^{a,1}$: the STD of a part (indexed by 1) in $\mathbf{O}_{by}^a$

$\sigma_{by}^{a,2}$: the STD of a part (indexed by 2) in $\mathbf{O}_{by}^a$

σ_{by}^{a,N_p} : the STD of a part (indexed by N_p) in $\mathbf{O}_{by}^a$

$\sigma_{int}^{a,1}$: the STD of a part (indexed by 1) in $\mathbf{O}_{int}^a$

$\sigma_{int}^{a,2}$: the STD of a part (indexed by 2) in $\mathbf{O}_{int}^a$

σ_{int}^{a,N_p} : the STD of a part (indexed by N_p) in $\mathbf{O}_{int}^a$

$\sigma_{o_\theta}^{a,1}$: the STD of a part (indexed by 1) in $\mathbf{O}_{o_\theta}^a$

$\sigma_{o_\theta}^{a,2}$: the STD of a part (indexed by 2) in $\mathbf{O}_{o_\theta}^a$

$\sigma_{o_\theta}^{a,N_p}$: the STD of a part (indexed by N_p) in $\mathbf{O}_{o_\theta}^a$

$\sigma_{rg}^{a,1}$: the STD of a part (indexed by 1) in $\mathbf{O}_{rg}^a$

$\sigma_{rg}^{a,2}$: the STD of a part (indexed by 2) in $\mathbf{O}_{rg}^a$

σ_{rg}^{a,N_p} : the STD of a part (indexed by N_p) in $\mathbf{O}_{rg}^a$

$\sigma_{x,n}^{a,1}, \sigma_{y,n}^{a,1}$: the STD of the spatial position of a control point (indexed by 1) along a contour instance (indexed by n)

$\sigma_{x,n}^{a,N_{cp}^n}, \sigma_{y,n}^{a,N_{cp}^n}$: the STD of the spatial position of a control point (indexed by N_{cp}^n) along a contour instance (indexed by n)

$\sigma_{by}^{s,1}$: the STD of conspicuity values in terms of blue-yellow of a part (indexed by 1) in $\mathbf{O}_{by}^s$

$\sigma_{by}^{s,2}$: the STD of conspicuity values in terms of blue-yellow of a part (indexed by 2) in $\mathbf{O}_{by}^s$

σ_{by}^{s,N_p} : the STD of conspicuity values in terms of blue-yellow of a part (indexed by N_p) in $\mathbf{O}_{by}^{s}$

$\sigma_{ct}^{s,1}$: the STD of the conspicuity values in terms of the contour feature over all control points along a contour instance (indexed by 1)

$\sigma_{ct}^{s,2}$: the STD of the conspicuity values in terms of the contour feature over all control points along a contour instance (indexed by 2)

$\sigma_{ct}^{s,N_{ct}}$: the STD of the conspicuity values in terms of the contour feature over all control points along a contour instance (indexed by N_{ct})

$\sigma_{int}^{s,1}$: the STD of conspicuity values in terms of intensity of a part (indexed by 1) in $\mathbf{O}_{int}^{s}$

$\sigma_{int}^{s,2}$: the STD of conspicuity values in terms of intensity of a part (indexed by 2) in $\mathbf{O}_{int}^{s}$

σ_{int}^{s,N_p} : the STD of conspicuity values in terms of intensity of a part (indexed by N_p) in $\mathbf{O}_{int}^{s}$

$\sigma_{o_\theta}^{s,1}$: the STD of conspicuity values in terms of orientation in θ of a part (indexed by 1) in $\mathbf{O}_{o_\theta}^{s}$

$\sigma_{o_\theta}^{s,2}$: the STD of conspicuity values in terms of orientation in θ of a part (indexed by 2) in $\mathbf{O}_{o_\theta}^{s}$

$\sigma_{o_\theta}^{s,N_p}$: the STD of conspicuity values in terms of orientation in θ of a part (indexed by N_p) in $\mathbf{O}_{o_\theta}^{s}$

$\sigma_{rg}^{s,1}$: the STD of conspicuity values in terms of red-green of a part (indexed by 1) in $\mathbf{O}_{rg}^{s}$

$\sigma_{rg}^{s,2}$: the STD of conspicuity values in terms of red-green of a part (indexed by 2) in $\mathbf{O}_{rg}^{s}$

σ_{rg}^{s,N_p} : the STD of conspicuity values in terms of red-green of a part (indexed by N_p) in $\mathbf{O}_{rg}^{s}$

σ_{υ} : the STD of observations of a measurement line of a predicted contour curve

$\hat{\sigma}_{k,f}^{l+1}$	:	a diagonal entry of the covariance matrix $\hat{\boldsymbol{\Sigma}}_k^{l+1}$
$\overline{\sigma}$	:	the STD of a probabilistic summary of a set of Gaussian distributions
$\overline{\sigma}_j^k$	:	an entry of the STD vector $\overline{\boldsymbol{\sigma}}_j^k$
$\boldsymbol{\sigma}_i^{j,k}$	:	the STD vector of the instance i of the part j in the local PNN of the object k or the STD vector of the control point i along the contour instance j in the global PNN of the object k
$\boldsymbol{\sigma}_{init}^{gb}$	:	the predefined initial STD vector of a new RBF in the updated global PNN
$\boldsymbol{\sigma}_{init}^{lc}$	:	the predefined initial STD vector of a new RBF in the updated local PNN
$\overline{\boldsymbol{\sigma}}_j^k$	:	the STD vector of $\overline{r}_j^k(\tilde{\mathbf{F}}_{lc})$ or the STD vector of $\overline{r}_j^k(\tilde{\mathbf{F}}_{gb})$
$\boldsymbol{\Sigma}_1$	:	the covariance matrix of one Gaussian distribution used to calculate Bhattacharyya distance
$\boldsymbol{\Sigma}_2$	:	the covariance matrix of the other Gaussian distribution used to calculate Bhattacharyya distance
$\boldsymbol{\Sigma}_{1,2}$	:	the combined covariance matrix of two Gaussian distributions used to calculate Bhattacharyya distance
$\boldsymbol{\Sigma}_i^{j,k}$	:	the covariance matrix of the instance i of the part j in the local PNN of the object k or the covariance matrix of the control point i along the contour instance j in the global PNN of the object k
$\boldsymbol{\Sigma}_p$	:	the covariance of prior measure uncertainty of the optical flow
$\boldsymbol{\Sigma}_\omega^m$	:	the covariance matrix of $\boldsymbol{\omega}^m$
$\boldsymbol{\Sigma}_\psi$	:	the covariance matrix of the conditional distribution of the optical flow
$\boldsymbol{\Sigma}_s$	:	the covariance of measure uncertainty of spatial derivatives
$\boldsymbol{\Sigma}_t$	:	the covariance of measure uncertainty of the temporal derivative

$\hat{\mathbf{\Sigma}}_i^l$	:	the covariance matrix of aggregate features of a vertex v_i at level l in an irregular pyramid
$\hat{\mathbf{\Sigma}}_{i,j}^l$	:	the combined covariance matrix of aggregate features of two vertices v_i and v_j at level l in an irregular pyramid
$\hat{\mathbf{\Sigma}}_j^l$	:	the covariance matrix of aggregate features of a vertex v_j at level l in an irregular pyramid
$\hat{\mathbf{\Sigma}}_k^{l+1}$	:	the covariance matrix of aggregate features of a vertex v_k at level $l+1$ in an irregular pyramid
$\overline{\mathbf{\Sigma}}_j^k$	:	the covariance matrix of $\overline{r}_j^k(\tilde{\mathbf{F}}_{lc})$ or the covariance matrix of $\overline{r}_j^k(\tilde{\mathbf{F}}_{gb})$
τ_a	:	the threshold used to determine if the orientation energy at a pixel is large enough
θ	:	an orientation
θ_d	:	the angular difference between the principal axis $\theta_{\mathbf{R}_g}$ of the proto-object and the principal axis θ_{ct} of the closed contour represented by $\mathbf{F}_{ct}^{t,\mu}$
$\theta_{\mathbf{R}_g}$	:	the direction of the principal axis of a proto-object $\mathbf{R}_g$ with respect to the Y-axis in the image coordinate system
$\theta_{\mathbf{R}_j^{attn}}$	:	the principal axis of $\mathbf{R}_j^{attn}$
$\wedge$	:	"logic and" operator
$\vee$	:	"logic or" operator

Chapter 1

Introduction

1.1 Motivation

1.1.1 Traditional Visual Perception

In the standard artificial intelligence literature [1], a robot agent is defined as something that perceives the external environment and acts on it, whose abstract model is shown in Figure 1.1. Perception and action are therefore two fundamental units for robots in the sense that the perception unit provides sensory information obtained from the environment, based on which the action unit produces corresponding behaviors by using the learned knowledge.

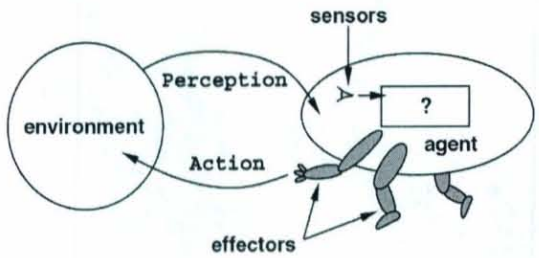


Figure 1.1: The abstract model of a standard robotic agent.

Thus designing a satisfactory perception system is the first important step for devel-

oping intelligent robots. Although a variety of sensing modalities, such as laser, sonar and audition, can be used for robots to perceive environmental information, vision is the primary sensory modality based on the fact that it has the ability to provide a large amount of information about the environment. This thesis therefore focuses on the research in the area of visual perception mechanisms used for robots.

However, every coin has two sides. Although the availability of a large amount of perceived data about the environment is a great advantage of visual perception in order to provide enough information for producing actions, it imposes a difficulty of how to effectively keep the balance between computing resources, time cost and fulfilling different visual tasks in the natural, cluttered and dynamic environment.

Organizing an effective representation of the environmental input is the key point to deal with this problem. There have been three categories of traditional methods proposed in the communities of computer vision and robotics: local feature based, local region based and global appearance based.

The local feature based methods attempt to represent the environment by detecting a set of features. These features are extracted using some specific methods, such as edge detector (e.g., Sobel operator [2] and Canny operator [3]), line detector [4], corner detector [5, 6] and scale-invariant feature transform (SIFT) [7]. These methods have been successfully applied to robotic tasks, such as localization [8], self-localization and mapping [9] and homing [10]. The advantage of this category of methods is that these local features are robust to scaling, lighting changes and so on. However, their main problem is that these extracted features are predefined by the programmers for a given special task or a special environment such that the robot has little flexibility and plasticity to observe new features when it is facing a new task or environment.

The local region based approaches segment a whole image into several regions and form a representation using those regions and their relationships. These methods have been mainly applied to robotic navigation [11–13]. The advantage of this category of methods is that an object region rather than a set of local spatial points is perceived,

which results in a higher perceptual resolution in the sense that more environmental information over the object region is available for visual perception. However, their main problem is the difficulty in effectively perceiving the object regions and adaptively forming the appearance representations of those local regions according to tasks and environments.

The global appearance based methods regard the input image as a whole and extract a profile that compactly summarizes the image's statistics and semantics. In these methods, several specific features, such as colors [14], textons [15], spatial envelope [16] and gist of the scene [17], are used as descriptors of the image and the histogram is employed for statistical analysis so as to create an overall profile of the image. These methods have been applied to robotic localization [14, 18] and navigation [19]. The great advantage of this category of methods is the capability for fast scene classification and localization. However, their main problem is the loss of local environmental information. The result is that they have difficulty producing precise actions, especially when there are local changes in the environment.

It can be seen that a common property of the above traditional perception methods is that perceptual behaviors are manually designed by programmers for a given task and environment. For instance, the edge feature is selected by the programmer for path tracking in the structured environment, whereas the SIFT feature is selected by them for localization in the cluttered environment. That is, the robot itself does not know what it is doing when it runs the perception program.

1.1.2 Cognitive Visual Perception: Selective Attention connects Perception to Action

The above discussions lead to a question: what type of perception system produces an intelligent robot? The intuitive idea is that an intelligent robot should have the mental capability of knowing how to perceive the environment autonomously. In other words, the intelligent perception system can give a robot the capability to explain what it is

doing during the perception process [20].

The best example of the intelligent perception system is human perception. Research on psychology and physiology [21, 22] has shown that a typical visual scene contains much information, not all of which can be fully processed by the visual system at a time. A selection mechanism is therefore employed by the human brain to filter out the irrelevant information. Selective attention [23–25] is such a mechanism, which serves to limit processing to one relevant item either in a conscious way according to the current task or in an unconscious way according to the present situation. In other words, the human brain knows how to perceive the environment autonomously by using the selective attention mechanism for perception. Since only the relevant part of the world is selected to be represented for action, perception and action can be linked through the process of selective attention [26, 27].

It can be seen that two aspects are required for this intelligent perception system.

- One is the conscious aspect, which can direct perception based on the task, context and knowledge learned from experience.
- The other is the unconscious aspect, which can direct perception in the case of facing an unexpected, unusual or surprise situation.

This selective attention based intelligent perception mechanism of humans is called ***cognitive visual perception*** in this thesis.

The objective of this thesis is therefore to develop a cognitive visual perception paradigm for robots, which can be used as the first step for further research on cognitive perception-action mapping.

1.2 Problem Statement

The proposed cognitive visual perception paradigm involves three successive stages: pre-attentive processing, attentional selection and post-attentive perception. The pre-attentive processing stage extracts pre-attentive features, based on which the attentional

selection stage mentally draws visual attention to an item at a moment. The objective of the post-attentive perception stage is to interpret the attended item in more detail in order to produce the correct action at the current moment and guide the further conscious attentional selection at the next moment. Thus this stage mainly includes perceptual completion processing, recognition and learning of the attended item.

This paradigm indicates that robotic visual perception starts from a low-level cognitive attentional selection procedure that guides attention to the relevant item of the scene, followed by a high-level post-attentive analysis procedure that analyzes the attended item and formulates it into an internal mental representation used for further cognitive behaviors. Since the attentional selection stage includes the conscious and unconscious ways, cognitive perception capability is realized by this stage. Therefore modeling the visual attention mechanism is the core part of the proposed perception paradigm.

Four fundamental problems will be solved in this thesis for modeling the proposed cognitive visual perception paradigm:

1. **Object-based Attention or Space-based Attention**

   There are two assumptions in psychological and physiological literatures attempting to understand the process of selective attention. The fundamental difference between them is the underlying unit of attentional selection. The space-based attention theory [23, 28–30] holds that attention is allocated to a spatial location. The object-based attention theory, however, posits that some pre-attentive processes serve to segment the field into discrete objects, followed by attention that deals with one object at a time [25, 31].

   In psychological and physiological communities, both theories can be supported [32–35]. However, in the area of computational research, object-based attention has the following advantages:

   - Object-based attention models are more robust than space-based attention models. This is because the attentional activation at the object level is esti-

mated by accumulating contributions of all continuous or discrete components within that object, whereas the activation at the spatial location level is estimated without accumulation. For instance, space-based attention has a higher possibility to be incorrectly attracted to a noisy location than object-based attention in a noisy scene, and space-based attention is more likely to be attracted to a wrong location than object-based attention in the case that objects overlap in a cluttered scene.

- Attending to an exact object can provide more useful information (e.g., shape and size) for robots to produce appropriate actions than attending to a spatial location.
- Object-based attention mechanism is the only way to realize top-down attention if the task-relevant feature is represented in terms of global features (e.g., shape).

Thus this thesis adopts the object-based visual perception idea [36] and the object-based visual attention theory [25]. That is, the world is pre-attentively parceled into objects, which are the underlying units for further attentional selection and post-attentive perception.

At present, most computational models of visual attention [37–41] focus on space-based attention. Little research [42,43] has been presented in modeling object-based attention. There are two main challenging problems about modeling object-based visual attention:

(a) Designing an effective and efficient pre-attentive segmentation algorithm;

(b) Estimation of object-based attentional activation.

2. **Modeling Conscious Attentional Selection**

Psychological and physiological research [24,25,28,44–49] has shown that there are two interactive manners which can direct attentional selection: bottom-up attention

and top-down attention.

Bottom-up attention guides attentional selection by means of the competition between each item and its spatial neighbors in terms of pre-attentive stimuli, which are extracted pre-attentively from the visual input. The salient and distinctive item that has competitive advantage over its neighbors can capture visual attention. Bottom-up attention can be regarded as an automatic and unconscious way to guide attentional selection since it captures attention based only on input stimuli without considering any task influences. Bottom-up attention has been successfully simulated and modeled in both psychological and computational attention models [37,38]. Bottom-up saliency of each item in the scene is estimated and then used to direct attention in these models.

It is important to note that the term *item* is generally used to represent the candidate unit of attentional selection, such as a spatial point and an object, in this thesis.

However, the purely bottom-up manner cannot guide attention to a desired item if the feature properties of that item are not unusual or salient [44]. Thus a top-down manner is also required. *Top-down attention* modulates the attentional selection based upon task instructions by biasing the pre-attentive stimuli preferred by the task. Therefore top-down attention can be seen as a conscious way to guide attentional selection.

Compared with the well-developed bottom-up attention models, how to build computational models of top-down attention is still a challenging problem. Some psychological models, such as guided search model (GSM) [44] and integrated competition (IC) hypothesis [49, 50], have been proposed as the basic theories to guide the modeling of top-down attention. Recent neurophysiological findings [51] have provided the neural response evidence to support the IC hypothesis: by directing attention to a task-relevant cue of an object, a competitive advantage over the

whole object is produced. This indicates that one type of task-relevant information can guide attentional selection to the desired object. It can be seen that the IC hypothesis not only summarizes a theory of the top-down attention mechanism but also integrates object-based attention theory. Therefore this thesis employs the IC hypothesis to model top-down attention.

Furthermore, four computational issues about modeling top-down attention by using the IC hypothesis will be solved in this thesis. Three of them are included in the attentional selection stage:

(a) Autonomous deduction of the task-relevant information given the task, context and learned knowledge;

(b) Estimation of the top-down biases based on the task-relevant information;

(c) Combination of top-down biases and bottom-up saliency.

The fourth computational issue is involved in the post-attentive perception stage:

(d) Learning of the attended object during the post-attentive perception stage.

3. **Development of Object Representations in Long-term Memory**

Object-based attention theory indicates that a general way of organizing the visual scene is to parcel it into discrete objects. These objects are the fundamental units, based on which perception and action are both performed. Thus the development of object representations in long-term memory (LTM), each of which is used as a carrier of the learned knowledge, is another key part of the proposed perception paradigm. The object representation in LTM can be seen as an internal mental representation of the object and it is termed as *LTM object representation* in this thesis. The term *development* represents two types of functions. The first function is constructing a uniform structure for the LTM object representations and the second function is learning the corresponding LTM object representation given the attended object at each moment.

Three issues will be solved for developing the LTM object representations in this thesis.

(a) Dual-function: The LTM object representation can be used to guide both top-down biasing during the process of perception and action selection during the process of action.

(b) Robustness: The developed LTM object representation can represent various instances of the object.

(c) Discriminability: The developed LTM object representation has the capability to discriminate itself from other objects, such that it can be effectively used for top-down biasing and action selection.

4. **Perceptual Uncertainty**

Sensory measurement is always subject to uncertainty. This perceptual uncertainty results in uncertainty in actions. A probabilistic approach is an alternative for dealing with uncertainty [52]. Thus, how to eliminate the uncertainty by integrating necessary probabilistic techniques during pre-attentive processing, attentional selection and post-attentive perception is an important issue.

1.3 Thesis Contributions

This thesis presents a cognitive visual perception paradigm for robots by solving the above challenging issues. This proposed perception paradigm has also been applied to robotic applications, including object detection and target tracking. A detailed discussion on the proposed work follows in the respective chapters and a summary of the main contributions of the thesis work is given below.

1. Presenting a framework of cognitive visual perception for robots using object-based attention mechanism: This framework divides visual perception into pre-attentive processing, attentional selection and post-attentive perception. As a result, robotic

visual perception starts from a low-level cognitive attentional selection procedure that guides attention to an object of the scene, followed by a high-level post-attentive analysis procedure that analyzes the attended object and formulates it into an internal mental representation used for further cognitive behaviors. The attentional selection stage supplies robots with the mental capability of knowing how to perceive the environment according to the current task and situation. Thus the proposed cognitive visual perception paradigm is adaptive and general to any task and environment.

2. Proposing a novel top-down attention method for object-based attention: Based on the IC hypothesis, this method uses one or a few conspicuous low-level feature(s) of the task-relevant object to guide attentional selection. These conspicuous low-level feature(s) can be autonomously deduced from the developed LTM representation of the target-relevant object. Meanwhile, the proposed method models top-down attention as a probabilistic procedure by using Bayes' rule and probabilistic estimation techniques. Thus this top-down attention method is more effective, efficient and robust than other methods and it is adaptive and general to any task and environment.

3. Proposing a probabilistic LTM object representation: The probabilistic neural network (PNN) [53] is used to construct the LTM object representation by probabilistically embodying various instances of that object. The result is that the learned representation can be used to direct top-down attention in the attentional selection stage, perform object recognition and learning in the post-attentive perception stage and guide action selection during the process of action. Dynamical learning algorithms are developed for training the LTM object representation.

4. Developing a pre-attentive segmentation algorithm: Pre-attentive segmentation is a demanding requirement to model object-based attention. This proposed algorithm divides the input scene into homogeneous proto-objects by extending irregu-

lar pyramid techniques [54, 55] and using a novel scale-invariant probabilistic similarity measure. This algorithm provides automatic, rapid and satisfactory results of pre-attentive segmentation for object-based visual attention.

5. Developing an effective, efficient and general method for object detection and target tracking by using this proposed cognitive visual perception paradigm: Two stages are used to model the processes of detection and tracking. The purpose of the attentional selection stage is to rapidly localize a candidate object by using either bottom-up attention or top-down attention; and the purpose of the following post-attentive stage is to validate the attended object by using high-level analysis.

1.4 Organization of the Thesis

The remainder of this thesis is structured into another eight chapters. Chapter 2 is concerned with the psychological and physiological background of visual attention, the state of the art of computational attention systems and robotic applications of visual attention, whereas the following six chapters present the details of the proposed cognitive visual perception paradigm.

Chapter 3 introduces the framework of the proposed cognitive visual perception paradigm. It gives an overview of the proposed perception paradigm.

Chapter 4 presents the pre-attentive processing stage. It includes pre-attentive feature extraction and a pre-attentive segmentation algorithm.

Chapter 5 presents the attentional selection stage. It involves a bottom-up attention model, a top-down attention model, the combination of bottom-up saliency and top-down biases, and the estimation of proto-object based attentional activation.

Chapter 6 presents the post-attentive perception stage. It includes perceptual completion processing, extraction of post-attentive features, development of LTM object representations and object recognition. Particular emphasis is placed on the development of LTM object representations.

Chapter 7 presents one robotic application of the proposed cognitive visual perception paradigm: object detection. This includes the detection of salient objects using bottom-up attention and the detection of task-relevant objects using top-down attention.

Chapter 8 presents another robotic application of the proposed cognitive visual perception paradigm: target tracking. Experimental results show that the proposed perception paradigm can achieve a satisfactory tracking performance to cope with difficulties, including changes in the background and the target, large variations of motion, partial and full occlusion and so on.

Chapter 9 finally states the conclusion of the thesis work and presents further research directions.

Chapter 2

Background on Visual Attention and Its Robotic Applications

2.1 Introduction

The visual attention mechanism is the core part of the proposed cognitive visual perception paradigm. This chapter aims to discuss the important background information related to the visual attention mechanism. Section 2.2 introduces some concepts and neurobiological findings of the human visual attention mechanism, which is the basis to model visual attention. Section 2.3 introduces several psychological models of visual attention, which are abstract theories attempting to explain the visual attention mechanism of humans and primates. Section 2.4 presents the state of the art of computational models of machine visual attention, which are built based on psychological models. Finally, section 2.5 surveys current advances on robotic applications of machine visual attention.

2.2 Concepts of Visual Attention

2.2.1 What is Visual Attention?

The term *attention* is widely used in our daily language and familiar to everyone. However, it is difficult to define and clarify visual attention precisely. It seems advisable to define visual attention from its intrinsic properties. Research in psychology and physiology has shown that visual attention has two basic aspects: limitation and selectivity [21, 22, 46, 56, 57]. *Limitation* means that the capability of processing information is limited in the brain, whereas *selectivity* represents the ability to filter out unwanted information. These two properties are interactive in the sense that either of them can be the reason for the other. If the limitation is regarded as the reason, visual attention can be defined as the mechanism that allocates limited visual resources for processing selected aspects of the visual input more fully than the non-selected aspects [58]. On the other hand, if the selectivity is regarded as the reason, visual attention can be defined as the mechanism that mentally selects one aspect from the visual input for processing according to the current task and situation [25,27]. This thesis adopts the second definition since it not only explains why visual attention is required (i.e., the processing resource is limited) but also clarifies how visual attention works (i.e., which aspects should be selected). The second definition can also be used to better explain why visual attention is the core mechanism for cognitive visual perception for robots in the sense that the visual attention mechanism gives robots the capability to know how perception works and what should be perceived.

2.2.2 Covert Attention and Overt Attention

Directing the focus of attention to an item of interest can be categorized into two ways [23,59–61]: *overt attention* (i.e., saccadic eye movement) that directs attention associated with eye movement, and *covert attention* that directs attention without eye movement. Some studies [59] have shown that covert attention and overt attention can work together.

That is, covert attention guides the focus of attention to an item of interest followed by a saccade that fixates the item and enables the perception with a higher resolution. On the other hand, other studies [60,61] have shown that either of them can work independently without the other.

An advantage of covert attention is that it is independent of motor commands [41]. The movement of eyes and head is not required to direct the focus of attention, with the result that covert attention is much faster than overt attention. Thus this thesis only studies the covert attention mechanism.

2.2.3 Space-based Attention and Object-based Attention

Whether attentional selection is space-based or object-based has been a controversial topic during the past decades [32,33]. The fundamental difference between them is the underlying unit of attentional selection.

Space-based attention theory holds that attention is allocated to a spatial location. For instance, the "spotlight theory" [23,28] proposes that attention is like a spotlight to illuminate the focused location and attention shifts along a path from one location to the next one, the "zoom-lens theory" [29] asserts that attention is covertly directed to a spatial region with the varying scope of its focus, and the "spatial gradient theory" [30] indicates that attentional selectivity is enhanced at a spatial location where the target stimulus is expected and the selectivity generally decreases with distance from that location. Numerous neurophysiological and neuropsychological experiments [62–67] have shown the space-based attention effects in the visual cortex.

In contrast, *object-based attention* theory posits that some pre-attentive processes serve to segment the visual field into discrete objects, followed by attention that deals with one object at a time [25,31]. Although the object-based theory is still in development, a number of useful findings from psychology [68–70] have been achieved. Furthermore, using the functional magnetic resonance imaging (fMRI) technique, neurophysiological experiments [51,71–73] have shown the object-based attention effects in the visual cortex.

Whether attentional selection is space-based or object-based is a controversial topic during past decades in psychological and physiological communities [32, 33]. Neurophysiological experiments have shown that attentional selection is space-based in some cases [62–67] and object-based in other cases [51, 71–73]. The above research has shown that space-based attention and object-based attention are not exclusive. In fact, they are reciprocal and intimated [34, 35]. The spatial location of an object can be treated as one of the various properties (e.g., color, shape and motion) of that object. The focus of attention, which is cued by one of these properties, is confined within the limits of the selected object.

Motivated by the object-based attention theory, an *object-based visual perception* idea has been furthermore exploited by recent psychological research [36]. Based on the experimental results of object-based attention, this idea holds that parsing the world into objects may occur quite early, and even pre-attentively. This idea challenges the traditional perception theories, which assert that perceptual systems do not parcel the world into objects and the organization of the perceived world into objects may be the central phenomenon of a human's thought systems [74].

Those findings indicate that object-based attention and perceptual organization must proceed together [25] and that they are reciprocal. More precisely, perceptual organization includes pre-attentive segmentation and post-attentive organization. Without pre-attentive segmentation, object-based attention will lose its selection ability. On the other hand, object-based attention can facilitate the post-attentive organization by sequentially putting the limited processing resource for the high-level perceptual analysis only on the attended object region at each moment.

Thus this thesis adopts the object-based visual perception idea and the object-based visual attention theory for modeling the proposed cognitive visual perception paradigm.

It is important to note that the term *object* used in the object-based attention theory can be best understood as the term *proto-object* that means the results of pre-attentive segmentation [36, 75]. The definition of proto-objects will be discussed in section 4.3.3 in

2.2.4 Bottom-up Attention and Top-down Attention

Directing the focus of attention can be initiated in two interactive mechanisms: bottom-up attention and top-down attention [44,46]. The *bottom-up attention* mechanism directs the focus of attention based on the conspicuousness of items in the spatial context, resulting that a salient item is selected to be attended. For instance, a white object will capture the bottom-up attention in the case that its neighbors are all black ones. Bottom-up attention can be seen as an innate perceptual behavior or as a behavior that can be developed gradually with experience [76]. In summary, bottom-up attention represents the unconscious aspect of perception, i.e., it is automatic and stimulus-driven.

An extensive concept of bottom-up attention can be described using the newly proposed term *surprise* [77–79]. Surprise is a mechanism that can attract the attention to an unusual or an unexpected item in both spatial and temporal contexts. In other words, it is referred to as both spatial conspicuousness and temporal novelty. This thesis only treats bottom-up attention as spatial conspicuousness.

On the other hand, the *top-down attention* mechanism directs the focus of attention based on the conscious instructions sent out from the brain. These conscious instructions are generated based on the knowledge, the current task and the context of mental states. For instance, if the task is to search for an orange, the item with orange color will attract the attention more easily than other items. In other words, top-down attention represents the conscious aspect of perception: The task, context and knowledge determine where you look [57], i.e., what you see is what you need [56]. Thus it can be said that top-down attention is conscious and task-driven.

Although understanding the top-down attention mechanism is still in development, three components which are responsible for guiding top-down attention have been clarified:

- Autonomous deduction of the task-relevant information given the task, context and

knowledge (e.g., the feature of the target) stored in LTM: Cognitive control [27,76, 80] is thought to be the mechanism to realize this component. Some psychological attention models, such as the IC hypothesis [49], have been proposed to model this component in detail.

- Estimate of the top-down biases based on the task-relevant information: Some psychological attention models, such as the biased competition (BC) hypothesis [46], have been proposed to model this factor in details.

- Combination of top-down biases and bottom-up saliency: Psychological research [44] has shown that bottom-up and top-down contributions are combined to decide which item is attended. The cognitive control mechanism is responsible for the combination.

2.2.5 Visual Cortices

Since visual attention is a concept of human perception, it is worthwhile to introduce some background information about visual cortices related to visual attention.

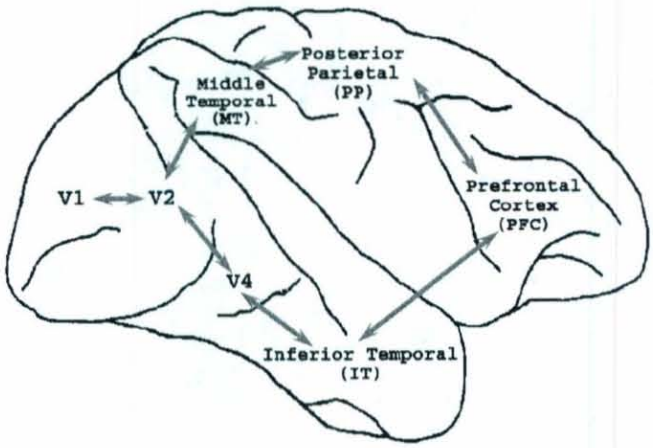


Figure 2.1: Visual cortices related to visual attention. Visual processing is divided functionally: the ventral stream (i.e., "what" pathway) leads to the inferior temporal cortex from V1, whereas the dorsal stream (i.e., "where" pathway) leads to the posterior parietal cortex from V1.

Visual cortices refer to the visual processing areas in the brains of primates and humans. The components related to the visual attention mechanism mainly include the primary visual cortex, the extrastriate cortical areas and the prefrontal cortex.

Primary Visual Cortex (V1)

The primary visual cortex is also known as striate cortex or V1. It is the most studied visual area in the brain. V1 receives information directly from the lateral geniculate nucleus (LGN). Area V1 has a well-defined map of the spatial information, i.e., the mapping from retina to area V1 is topographical in that nearby regions on the retina project to nearby regions in V1.

The main function of area V1 is analogous to local spatiotemporal energy filters. In other words, area V1 is associated with the neuronal processing of spatial frequency and local orientation energy [58]. These orientation-sensitive cells in area V1 can be grouped into three functional classes: simple cells, complex cells, and hypercomplex cells [81]. The receptive fields of simple cells are sensitive to lines and step edges as well as orientations of them. Even-symmetric filters (i.e., line detectors) and odd-symmetric filters (i.e., step edge detectors) can be used to model simple cells. Complex cells respond to more complex patterns, such as specific orientations and directions of movement without any phase information. Thus, complex cells can be modeled by summing the outputs of a group of simple cells with similar orientations. Hypercomplex cells, also called endstopped cells, exhibit end inhibition so as to localize line-ends and corners. Based on the fact that the excitatory influence from the small receptive field and the inhibitory influence from the large receptive field converge in the hypercomplex cell [82], the model of hypercomplex cells can be achieved by using the difference between the responses of two complex cells at the same central position and orientation, but of different receptive size.

Extrastriate Cortical Areas and Visual Pathways

The extrastriate cortical areas include V2, V4, inferior temporal cortex (IT), middle temporal area (MT or V5) and posterior parietal cortex (PP).

These visual processing areas appear to be organized as two major pathways or streams: ventral stream and dorsal stream, as shown in Figure 2.1. Along each pathway, the complexity of visual processing increases and the receptive field size of an individual neuron increases.

- **Ventral stream:** It begins with V1, goes through areas V2 and V4, and goes on to the IT cortex [83]. This stream is responsible for object recognition and thereby it is also named as the "what" stream. In this stream, area V2 responds to illusory or subjective contours [84], area V4 mainly responds to colors [85] and the IT cortex responds to the complex object features, such as shapes [86].

- **Dorsal stream:** It begins with V1, goes through area V2 and area MT, and goes on to the PP cortex. It is associated with motion processing and location representations, and thereby it is also named as the "where" pathway. Motion and depth are processed in this pathway [87]. For example, area MT plays a role in perception of motion [66].

Neurobiological studies have shown that a single cortical area can not successfully guide attention. That is, attentional selection is correlated with nearly all visual cortices [88], including the IT cortex [63,89], the PP cortex [62], area MT [66], area V4, area V2 and area V1 [64,67,71].

Prefrontal Cortex

The prefrontal cortex (PFC) is the anterior part of the frontal lobes of the brain, lying in front of the motor area. Neurons in the PFC encode many different types of information at all stages of the perception-action cycle. The PFC provides the neural basis of the *cognitive control* mechanism [27,76,80], whose function is to plan and control external and

internal behaviors according to the task, situation and perception-action rules. External behaviors include the actions executed by the external effectors, whereas internal behaviors include the attentional selection. The PFC exhibits the following neural properties related to cognitive control:

- **Development of Perception-action Rules:** It develops the perception-action rules, which regulate two types of associations [27,80,90,91]: One is the association between attentional states and external behaviors, and the other is the association between the current attentional state and the conscious prediction of the next attentional state.

- **Memory retrieval:** The PFC has the ability to recall the corresponding perception-action rule and the representations of the task-relevant object from LTM [27].

- **Top-down selectivity:** The PFC plays a central role to direct top-down attention [46,92], including deduction of the task-relevant information as well as combination of autonomous bottom-up attention and conscious top-down attention.

2.3 Psychological Models of Visual Attention

Modeling visual attention can be categorized into two levels: psychological and computational. The objective of psychological modeling is to simulate data from behavioral and neurophysiological experiments. Psychological models include descriptive ones [24, 25, 28–30, 44, 46–48] that summarize the basic theories and principles of attention, and computational ones [93] that are used to compare with experimental data. On the basis of psychological models, biologically-plausible computational models can be built for applications in computer vision and robotics. Following subsections describe four psychological models of visual attention in detail since they have been widely used as the fundamental theories for computationally modeling visual attention.

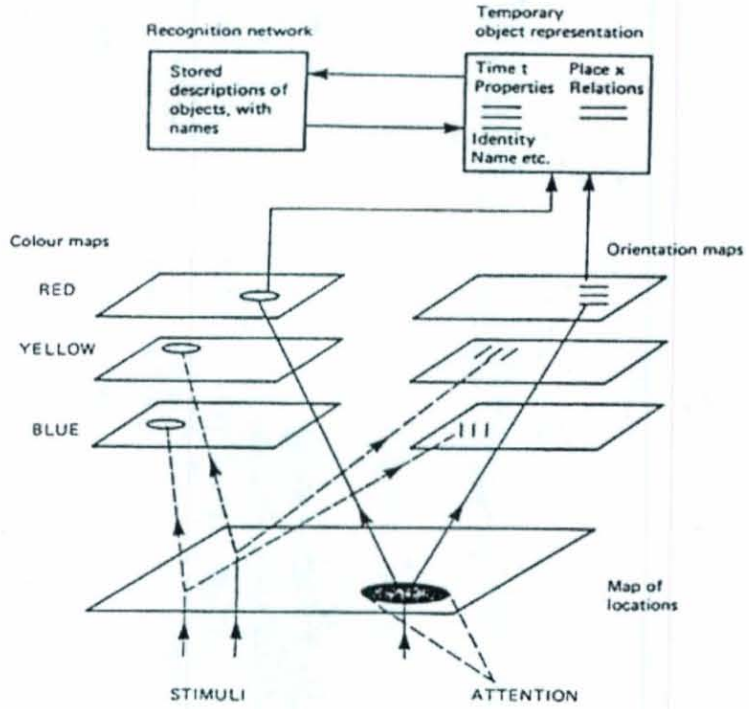


Figure 2.2: Feature integration theory (FIT). The visual scene is initially coded in parallel along a variety of feature dimensions, such as brightness, color, orientation and direction of movement. All features are combined together spatially to achieve a location-based master map that exhibits the saliency of each location, and attention is focused on a salient location by scanning that master map. The figure is from [94].

2.3.1 Feature Integration Theory

The feature integration theory (FIT) [24], proposed by Treisman in 1980, is one of the early and highly regarded theories in the field of visual attention. It is graphically illustrated in Figure 2.2. The basic idea of FIT is that features, rather than a unitary object as claimed by Gestalt psychologists, come first in perception. In this model, features are registered early, automatically and in a parallel mode across the visual field, while objects are identified separately and only at a later stage, which requires focused attention [24].

This theory characterizes two properties of visual attention:

- The visual scene is initially coded in parallel along a variety of feature dimensions, such as brightness, color, orientation and direction of movement.

- Focused attention provides a way to integrate the initially separated features into a whole object. That is, locations of these separated stimuli are processed serially with focused attention. Any features that are present in the central fixation of attention are combined to form an object for further perception. Without focused attention, features cannot be related to each other.

FIT was further developed by adapting recent research findings [94, 95], in which a detailed framework of attention was presented to elucidate several important aspects about how the focus of attention is directed.

- Each feature dimension consists of several feature maps. For example, color dimension is composed of red, green, blue and yellow.

- Location information is coded in the feature maps. That is, feature maps are topographically organized.

- Attentional competition performs in a location-based serial manner. That is, all features are combined together spatially to achieve a location-based master map that exhibits the saliency of each location, and attention is focused on a salient location by scanning that master map.

It can be seen that the main contribution of FIT is that it provides a basic framework of the bottom-up attention mechanism. A spatial location (extensively called an object) can be detected very quickly in the case that it differs from its spatial neighbors in terms of one or more features.

It should be noted that FIT uses the term *feature dimension* to represent a separated feature domain and uses the term *feature map* to represent a discrete category in a feature dimension. For instance, color and orientation are regarded as feature dimensions; red feature maps, green feature maps, blue feature maps and yellow feature maps are regarded as feature maps in the color dimension.

In summary, FIT can be seen as a psychological model of bottom-up, space-based attention.

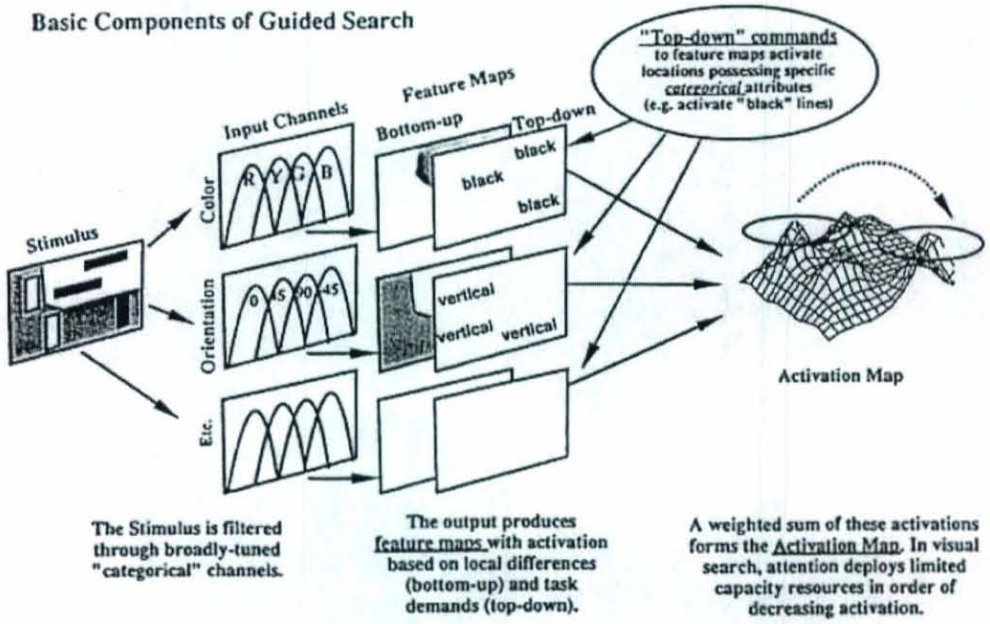


Figure 2.3: Guided search model (GSM). The strength of bottom-up activation for one location is based on the differences between it and items at neighboring locations in terms of feature maps. The top-down request for a given feature will activate the locations that might contain that feature. An overall activation map is created by a weighted sum of all top-down and bottom-up activations. The figure is from [44].

2.3.2 Guided Search Model

Another theory, named guided search model (GSM), was proposed by Wolfe [44] to model the top-down attention mechanism in conjunction with bottom-up attention. It is graphically illustrated in Figure 2.3.

GSM is built by extending FIT and it asserts that human perception, such as search behavior, can be divided into two stages. A pre-attentive stage carries out a parallel information processing in terms of basic features (e.g., color, motion and depth) across the whole visual field, followed by a post-attentive stage that performs further complex operations (e.g., object recognition) over a limited portion of the visual field with the guidance of visual attention.

It is important to note that GSM uses the term *activation*, rather than *saliency*, to

represent the attentional strength obtained by a combination of bottom-up and top-down attention, since saliency can only represent the bottom-up attentional strength.

GSM asserts that the role of the pre-attentive stage is to identify locations that are worthy of further attention. Similar to FIT, GSM proposes that in the pre-attentive stage the input stimuli are in parallel separated into several independent topographical feature maps along each feature domain, such as a "red" map in the color domain or a "vertical" map in the orientation domain. In particular, GSM adopts the idea that the orientation domain consists of broadly tuned categories (i.e., feature maps), which respond to steep, shallow, left or right.

GSM proposes that the strength of bottom-up activation is dependent on the differences between a location and its neighbors. It codes the differential activation of locations in each feature map. In other words, the activation of bottom-up attention for one location is based on the differences between that location and its neighbors in the relative feature maps.

The most important contribution of GSM is to model top-down attention. It posits that a top-down request for a given feature will activate the locations that might contain that feature. In each feature map, the top-down activation of a location is determined by its match to the corresponding properties of task-specific targets.

Furthermore, GSM claims that the combination of contributions of different feature maps and the combination of top-down and bottom-up activations can be modulated by the cognitive control mechanism. However, it does not give a method to model the modulation process.

Finally GSM asserts that an overall activation map can be created by a weighted sum of all top-down and bottom-up activations. This map is used to guide attentional selection in the sense that the location with the greatest activation is selected to be attended.

In summary, GSM can be seen as a psychological model of space-based attention by the combination of bottom-up and top-down attention.

2.3.3 Biased Competition Hypothesis

The biased competition (BC) hypothesis [46] was proposed by Decimone and Duncan in 1995. It is based on two aspects of visual attention: selectivity and limitation, as have been discussed in section 2.2.1.

Unlike FIT and GSM, which assert that attentional selection is a combination process based on activations along feature maps, the BC hypothesis posits that attentional selection is a biased competition process. Competition is biased in part by the bottom-up mechanism that separate items from their background and in part by the top-down mechanism that favors items relative to the current task.

Bottom-up bias competition is based on two neural mechanisms. One is the competition in the spatial context. The responses of many neurons to an optimal stimulus within their classically defined receptive field may be completely suppressed if similar stimuli are located within a large surrounding region. This results in the biasing towards local inhomogeneities, i.e., items that are larger, brighter, faster moving and so on. The other is the competition in the temporal context. This indicates the case of biasing to novelty. In the temporal domain, stimuli stored in memory may function as the temporal surround against which the present stimulus is compared. The temporal context of a stimulus may contribute as much to its saliency as its spatial context.

BC hypothesis also holds that the neural mechanism underlying the selection of top-down specification requires a means to hold the sought-after item in working memory (WM) and uses this memory to resolve competition among the items in the scene. This top-down bias competition is modeled as follows. According to the task, an *attentional template* in WM is formed to represent the short-term description of the task-relevant information currently needed (e.g., an object with a certain feature or in a certain location), such that input stimuli matching that attentional template are favored in the visual cortex. BC hypothesis further proposes that top-down biasing performs not only in the feature domain (e.g., shape, color and size) but also in the spatial domain (i.e., location).

Thus the idea of an attentional template is one of the contributions of the BC hypothesis. It regulates the way to estimate top-down biases given the task-relevant information. However, how to deduce the task-relevant information is not considered in the BC hypothesis.

In addition, the BC hypothesis holds that features and locations might be linked to some extent within the ventral stream, which supports the idea of constructing topographical feature maps proposed by FIT and GSM.

In summary, the BC hypothesis can be seen as a psychological model attempting to explain the visual attention mechanism, regardless of being space-based or object-based, through the novel view of biased competition.

2.3.4 Integrated Competition Hypothesis

By extending the BC hypothesis, Duncan et al. [49, 50] further presented the integrated competition (IC) hypothesis to explain the object-based attention mechanism.

IC hypothesis regulates the object-based attentional selection by using the following three general principles.

- **Competition:** In most of the neural systems activated by visual input, processing is competitive in the sense that enhanced response to one object is associated with decreased response to others. In other words, responses to different objects may be mutually inhibitory.

- **Top-down biasing:** Top-down priming of neural activity biases the competition towards objects of relevance to the current task. Furthermore, this selective priming shows flexibility in top-down attentional control. That is, any kind of visual properties can be task-relevant information used to direct attention and assign limited processing capacity to the task-relevant object. These properties include color, size, shape, direction of motion, location and so on. Each property is processed in its own neural system.

- **Integration:** The competition is finally integrated between components of the sensorimotor work. As an object gains dominance in terms of any one property (i.e., relevant property), responses to this same object are enhanced elsewhere (i.e., irrelevant properties). In this way the numerous properties of the object are made concurrently available for subsequent attention processing. In other words, one or some of the task-relevant features can activate the whole object including relevant and irrelevant properties to be attended.

One contribution of the IC hypothesis is to elucidate how object-based attention works by using the integration principle. This principle supports the object-based attention theory [25], i.e., the unit of attentional competition is an object. Once any property of an object successfully captures the attention, the other properties of that object can be integrated to form a complete object to be attended. This means that the construction of object representations in WM from the conjunction of many different features appears to occur in parallel before individual objects are selected for attention.

The other contribution of the IC hypothesis is the top-down biasing principle, which asserts that any property of an object can be used as task-relevant information to guide the top-down attention. The task-relevant property can be either deduced from the intrinsic properties of the task-specific object or directly specified by the task.

It is important to note that the IC hypothesis has been supported by recent neurophysiological research. The results of some neurophysiological experiments reported recently [51, 71] are in accord with the IC hypothesis.

In summary, the IC hypothesis can be seen as a psychological model of object-based attention with special emphasis on regulating top-down attention.

2.4 Computational Models of Visual Attention

Based on psychological models, several computational models of visual attention have been proposed. This section introduces some of the most important computational mod-

els in the areas of computer vision and robotics.

2.4.1 Koch's Model

The first approach to the computational architecture of visual attention was introduced by Koch and Ullman [37]. It is inspired from FIT. The key point of this architecture is that several features are computed in parallel and their conspicuity is collected in a saliency map. This architecture further presents a Winner-take-all (WTA) network to determine the most salient region in the topographic saliency map. WTA is implemented by the artificial neural network, in which synaptic interactions among units ensure that only the most active location remains whereas all other locations are suppressed. Thus WTA is a biologically-plausible approach to selection of a maximum. This architecture also suggests an inhibition of return (IOR) mechanism for inhibiting the selected region. IOR causes an automatic shift towards the next most salient location.

2.4.2 Itti's Model

One of the popularly known computational attention systems is Itt's model [38], which serves as a basis for many research groups. It is derived from Koch' model. This model characterizes space-based bottom-up attention as proposed in FIT. Figure 2.4 shows the framework of Itti's model.

The most important contribution of Itti's model is that it provides a complete method for modeling the space-based bottom-up attention mechanism. It basically includes two operators: center-surround difference and across-scale combination.

In Itti's model, three dimensions of pre-attentive features are first extracted from the visual input. Those dimensions include brightness, colors and local orientations. Itti's model employs the center-surround difference values, rather than the absolute feature values proposed by FIT, to construct a set of multi-scale feature maps. This is on the basis of two kinds of neural mechanisms. One is that intensity contrast is detected by neurons sensitive either to dark centers on bright surrounds or to bright centers on dark

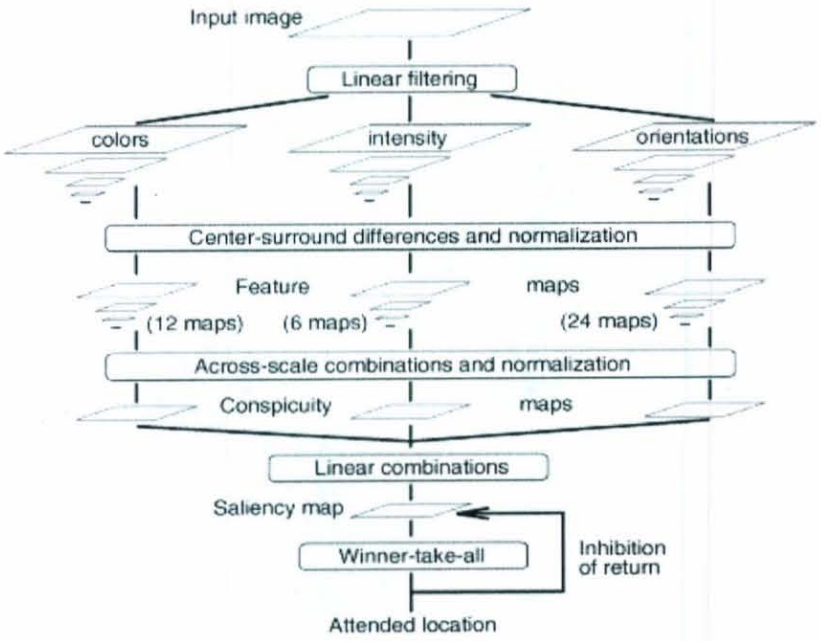


Figure 2.4: General architecture of Itti's model. From the visual input, three dimensions of features are extracted: intensity, colors and local orientations. A center-surround difference operator is then used to model the bottom-up competition mechanism and achieve a set of multi-scale feature maps. Those feature maps are finally combined to yield a location-based master saliency map by using an across-scale combination operator. This figure is from [38].

surrounds [38]. The other is that color dimension is represented by using a so-called "color double-opponent" system in the cortex [96]: in the center of their receptive fields, neurons are excited by one color and inhibited by another, while the converse is true in the surround. Such spatial and chromatic opponency exists for the red-green and blue-yellow color pairs in the human primary visual cortex. Accordingly, Itti's model creates a Gaussian pyramid at nine spatial scales for each feature category, including intensity, red-green pair, blue-yellow pair and orientation with four preferred directions. The center-surround differences are achieved by calculating the difference between upper and lower scales in the Gaussian pyramid of a feature category. It can be seen that Itti's model uses the center-surround difference operator to simulate bottom-up attentional competition, based on the fact that the neural mechanism of bottom-up competition is

the contrast between one location and its neighbors in terms of pre-attentive features so as to highlight the location which has competitive advantage over its neighbors.

In order to simulate the feature integration process as indicated by FIT, Itti's model normalizes all feature maps into a fixed value range and then combines them in a master saliency map by using an across-scale combination operator. The master saliency map is location-based. Finally, spatial locations compete for attention based on the master saliency map such that only locations which locally stand out from their surround can persist. The WTA network is used based on the master saliency map to find out the most salient location, which is the focus of attention.

2.4.3 Navalpakkam's Model

Itti's model is a basic version that concentrates on computing bottom-up attention. The need for top-down influences is mentioned but not realized in Itti's model. In recent research, Navalpakkam and Itti [39] introduced a derivative of Itti's model in order to deal with the task-specific guidance of visual attention. This derivative version is called Navalpakkam's model in this thesis. It can be seen as a space-based attention model by combining top-down and bottom-up attention.

Navalpakkam's model characterizes top-down attention as follows. Given a task specification in the form of keywords, the task-relevant entity is determined and the learned representation of this entity is recalled from LTM. This representation is then used to bias the pre-attentive features so as to guide the attention towards an instance of the task-relevant entity in the present scene.

One contribution of Navalpakkam's model is the approach to deducing the task-relevant objects (termed as task-relevant entities in Navalpakkam's model) given a task by using the knowledge representation techniques in the area of artificial intelligence. A symbolic LTM is built to act as a knowledge database, including entities and their relationships. Given a task specification, the task-relevant entities are determined by using the inference algorithm in the knowledge database.

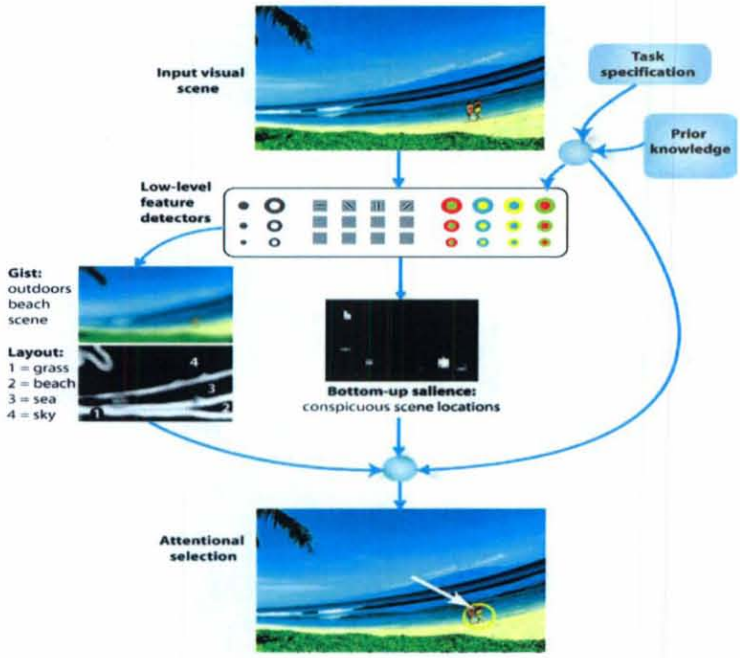


Figure 2.5: General architecture of Navalpakkam's model. It is a space-based attention model by combining top-down and bottom-up attention. Given the task, the task-relevant entity is first deduced based on a knowledge inference algorithm. The representation of that task-relevant entity is then recalled from LTM, based on which top-down biases are estimated. Top-down biases are integrated into the corresponding feature maps and a location-based attentional guidance map is finally produced to direct the focus of attention. This figure is from [39].

Another contribution of this model is to build a multi-scale object representation in LTM as the prior knowledge of the task-relevant entity. Once the task-relevant entity is deduced, the prior knowledge of that entity is recalled from LTM and temporarily stored in WM so as to bias the corresponding pre-attentive features. The object representation is learnable. When attention is directed to a location in that object, a 42-component vector consisting of center-surround difference features at multiple spatial scales is obtained around that attended location. By combination of those vectors obtained at different views, the probabilistic distribution of this vector is produced and stored as the object representation in LTM. However, one disadvantage is that the object representation cannot represent the object precisely when that object consists of multiple parts since it is

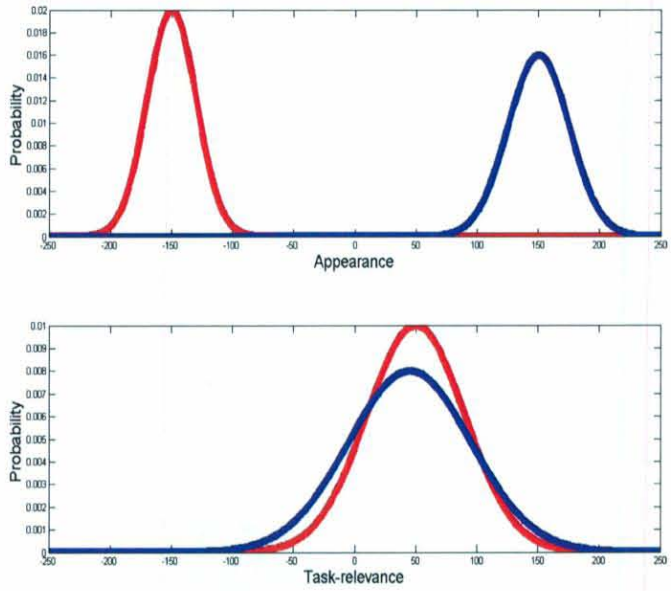


Figure 2.6: One shortcoming of the top-down biasing method in Navalpakkam's model. In this case, the target and a distractor have the similar task-relevance, but have different appearance values in a feature dimension. As a result, the top-down biasing of Navalpakkam's model cannot effectively work since it only uses the task-relevance. The red curves represent the distributions of appearance and task-relevance of the target, while the blue curves represent the distributions of appearance and task-relevance of a distractor in the same scene.

only learned at one salient location in that object.

Once the object representation of the task-relevant entity is recalled from LTM, the top-down bias of each feature map is estimated by using the feature relevance, which is computed based on the distribution parameters of that feature in the learned object representation. Finally, Navalpakkam's model multiplies each feature map with the corresponding top-down bias to yield a location-based attentional guidance map, based on which the WTA network is used to find out the most salient location as the focus of attention. However, this top-down biasing method might be ineffective in the case that the environment contains distractors which share the relevance with the target in terms of some features. This case is illustrated in Figure 2.6, in which the target and a dis-

tractor have the similar task-relevance, but have different appearance values in a feature dimension. As a result, the top-down biasing of Navalpakkam's model cannot effectively work since it only uses the task-relevance.

A noticeable point of this model is that the gist information is integrated as a top-down cue. Gist information represents what observers can gather from a scene with a single glance. It can be regarded as a relatively low-level scene representation which is acquired over very short time frames. It is likely to indicate the target location information in top-down attention. For example, if the task is to find humans and the gist is an outdoor beach scene, humans could be found by focusing attention near the water and the sand.

2.4.4 Other Space-based Attention Models

Frintrop's Model

Frintrop [41] presented a computational system for space-based visual attention, which is called VOCUS. This system extends and improves Itti's model in several aspects, ranging from implementation details to conceptual revisions, such that this system enables a considerable gain in performance and robustness.

One contribution of this system is the top-down attention method for target detection. In the learning phase, the system learns the target's features, which are the feature maps computed by using Itti's model, and then computes the top-down bias weights for all feature dimensions. In the search phase, the learned bias weights of the target are used to bias the corresponding feature maps. This top-down attention method is similar to the one in Navalpakkam's model, except for the computational method of bias weights.

The other contribution of this system is the extension of the attention model to different sensor modules, such as the laser scanner.

Hamker's Model

Hamker [40] proposed a space-based attentional system to model the visual attention mechanism of the human brain. Although the main objective of this model is to explain human visual perception and gain insight into its functioning, it also provides a computational method used for a machine's visual attention.

Hamker's model simulates bottom-up attention by sharing several aspects with the architecture of Itti's model. It computes the contrasts in terms of several feature dimensions, including intensity, red-green pair, blue-yellow pair and local orientations. Those contrasts are then combined into a perceptual map (i.e., master salience map).

In addition to the bottom-up behavior, Hamker's model also simulates the top-down influences. This model learns the target by remembering the feature values of that target, which is placed in a black background so as to exclude the background disturbance. Those learned features are then memorized in WM in order to influence the conspicuity of the features in the present scene. It finally merges the conspicuities of bottom-up and top-down cues to direct attention.

Begum's Model

Begum et al. [97–99] presented a probabilistic approach to modeling visual attention. The most important contribution of this model is to regulate visual attention as a Bayesian inference process. It uses a Bayesian filter and dynamically constructed Gaussian adaptive resonance theory to recursively estimate the focus of attention.

2.4.5 Sun's Model

All the above reviewed computational models are for space-based attention. In contrast, Sun and Fisher [42] presented a sophisticated framework for computing object-based attention, which is the first computational model of object-based visual attention.

This model posits that the salience of a perceptual grouping is a function of all salience

contributions emerging from the components within that grouping. The component represents a spatial point within a perceptual grouping. Those salience contributions work together to compete with their common competitors (i.e., other perceptual groupings) and compete with each other. In Sun's model, the salience of a component is estimated by using Itti's model. That is, the salience of a component is calculated by combining the space-based conspicuity of that component in terms of intensity, colors and orientations.

Sun's model guides attentional selection based on the grouping based salience, i.e., perceptual groupings are the basic units of attentional selection. At any given moment, enhanced responses to one grouping will decrease responses to other competitive groupings. Once one grouping wins the dominance of selective attention, all other relevant processing to this grouping and all components belonging to this grouping share the same dominance.

In Sun's model, the perceptual grouping is in a hierarchical structure. In this sense, a grouping can be a point, a region, a feature, an object, a group of features or a group of objects. Accordingly attentional selection is also hierarchical in the sense that attention can select a location, a feature, a discrete object or a group of objects. Thus this model integrates space-based attention and object-based attention into a uniform framework, where space-based and object-based attentional selectivity are either cooperative or independent of each other for efficient selection according to the current visual situations and tasks.

However, there remain two problems in Sun's model. One is how to obtain the perceptual grouping pre-attentively. The groupings used in Sun's model are manually created. The other is how to model top-down attention for the object based attention.

2.4.6 Other Object-based Attention Models

Orabona's Model

Orabona et al. [43] presented another object-based attention model for humanoid robots. In Orabona's model, the salience evaluation is based on the psychological idea of proto-objects, which ares defined as blobs of uniform color in the image. A watershed transform algorithm is employed to implement the pre-attentive segmentation based on uniform color to produce a set of perceptual blobs.

By training the humanoid robot to learn the object at different views, an internal representation of that object is formed. Such representation is a vector consisting of color opponent features (i.e., red-green and blue-yellow pairs). This representation provides the top-down cues to bias attention towards the task-relevant target. The top-down biases are calculated as the Euclidean distance in the color opponent space between each proto-object's color in the real scene and the average color in the target representation.

Aziz's Model

Aziz et al. [100] proposed another object-based visual attention model, which is promising for real-time applications for robots. One contribution of Aziz's model is the integration of perceptual segmentation into the attention model. This model introduces that hue-intensity-saturation (HIS) color space is an appropriate representation of human color perception [101]. It employs a region growing algorithm for perceptual segmentation in the HIS color space. The resulting segments are then used to compute features via their convex hulls.

Although the pre-attentive segmentation algorithms are proposed in both Orabona's model and Aziz's model, there are two limitations of those pre-attentive segmentation algorithms. The first limitation is that only color features are integrated such that a variety of other features (e.g., intensity and orientations) are lost and those algorithms cannot be used for an intensity image. The second limitation is that those algorithms

are not robust for various conditions, such as noisy and outdoor environments.

2.5 Robotic Applications of Visual Attention

This is an emerging and highly interesting topic of applying the visual attention mechanism to robots in recent years. This section categorizes current robotic applications of visual attention into two groups. One is for a single and specific robotic task, such as object detection, target tracking, localization and navigation. The other is for general robotic perception. Also, the robotic applications of visual attention can be categorized as applying space-based attention or object-based attention, and applying bottom-up attention or top-down attention. This section first reviews several applications of single robotic tasks and then reviews a few applications of general robotic perception.

2.5.1 Object Detection and Recognition

The most common application of visual attention is the object recognition task [102]. The attention mechanism can divide object recognition into a two-stage procedure: the pre-attentive stage selects a candidate object to be attended, and then the post-attentive stage functions as a classifying recognizer on the attended candidate object [102]. In the area of robotics, object detection and recognition is also an important ability for completing more complex tasks.

Although some example applications in object recognition have been proposed [103, 104], the most popular application of visual attention into object recognition is proposed by Walther et al. [105]. In this system, Itti's model [38] is used to select one object of interest in the bottom-up attention way and then the SIFT [7] feature is employed to recognize the attended object. The SIFT feature is a set of high-dimensional descriptors at the extracted keypoints. These descriptors have the invariance to scaling, rotation, spatial transformation and illuminative effects, such that they have a great advantage for recognition. However, the SIFT feature is computational expensive for setting up

the correlation during the process of recognition. Therefore, the integration of attention can greatly reduce the computational cost since only an attended object, rather than the entire image, is used for recognition.

The problem of all those proposed applications is that only bottom-up attention is employed, with a result that the attended object is not the target to be recognized in most cases.

2.5.2 Target Tracking

Target tracking in a dynamic environment is an important ability of robots in many tasks such as surveillance. Some visual attention based tracking methods have been proposed, such as [106], where the salient locations are tracked over time. One problem of current visual attention based tracking methods is that only space-based bottom-up attention is used, so that these methods fail in the case that the target is not salient. The other problem is that the complete target region cannot be tracked.

2.5.3 Localization

Visual attention, especially bottom-up visual attention, has a great potential for landmark detection in applications of robotic localization due to the selectivity of attention. The focus of attention highlights a limited number of possible items of interest in an image, which provide the clues to select the landmarks. Especially in the outdoor environment and open areas, the standard methods for localization, like matching 2-D laser range and sonar scans, are likely to fail. Thus selective attention is a promising alternative to those cases.

Bottom-up attention for self-localization

A self-localization method for robots based on space-based bottom-up attention has been proposed in [107]. Basically this method employs the bottom-up attention mechanism to

locate the salient and reliable locations and uses the features at that location to form a landmark representation. During the learning phase, the bottom-up attention model detects the most salient features along the robot path. After characterizing them by using a visual descriptor vector, this set of salient features together with location information become the candidate landmarks. By tracking the detected features over time, the robot evaluates their robustness and selects the robust candidates as the landmarks followed by organizing the selected landmarks into a topographic map. The self-localization is realized by matching the detected features in the current scene to the learned representation of stored landmarks so as to determine the position of the robot.

Bottom-up attention for SLAM

A simultaneous localization and mapping (SLAM) method using space-based bottom-up attention has been proposed by [108]. The contribution of this method is to use the salient locations to form a topographic map. Thus this map has the capacity to predict the position and appearance of landmarks. Comparison of the prediction with the current observation allows redetection in loop closing situations.

Another method [109] is also proposed for a robot's environmental exploration. This method estimates a bottom-up saliency map and integrates SLAM metric information into that map.

Combination of attention and gist for localization

A new method of combining the attention mechanism and gist information for localization was proposed by [17]. Besides the selectivity of attention which is useful for landmark detection, this method found that human vision exhibits the ability to rapidly summarize the gist of a scene. Attention and gist are contrasting and complementary: selective attention prefers local and detailed information, whereas gist gives global and coarse information. Both of them rely upon raw features that come from the same area, the early visual cortex. This method uses the gist information as a place (scene) recog-

nition. During the training phase, the landmarks detected by the bottom-up attention mechanism are clustered in terms of gist information. During the localization phase, gist information can enable the system to prioritize the on-line landmark search process.

2.5.4 Navigation

Current applications of the visual attention mechanism in robotic navigation are limited to some simple tasks and environments. In [110] a mobile robot uses an attention system for navigation. In this application, the robot is guided towards a large object by its visual attention based on the fact that the large object has great salience in the scene.

Another application was presented in [111]. An attention system is used to support autonomous road following by highlighting the relevant regions in a saliency map.

It is obvious that those proposed methods basically use bottom-up attention. There is potential in further research to use top-down attention together with bottom-up attention for navigation applications. Top-down attention can decide what to be attended according to the current task (i.e., plan) and situation. For example, the top-down attention can direct perception to pedestrians for avoidance when the robot is passing through a street intersection.

2.5.5 General Visual Perception for Robots

As we know so far, only one approach has been proposed in [112] to model robotic perception by using the visual attention mechanism. This approach models robotic perception as a two-stage procedure: pre-attentive processing and post-attentive processing. The attention mechanism guided by the present behavior is modeled in the pre-attentive stage.

One contribution of this approach is the top-down attention mechanism simulated by deducing the behavior-relevant feature dimension based on both perceptual factors and motivational factors and giving more weight to the relevant feature dimension. However,

this approach does not estimate the location-based top-down biases as posited by GSM [44]. It is important to note that this approach employs space-based attention.

2.5.6 Mapping between Perception and Actions

A general paradigm of mapping between perception and actions for developmental robots is proposed by Weng et al. [113,114]. This paradigm is called autonomous mental development (AMD). The basic objective of AMD is to construct a cognitive mapping mechanism between perception and actions, with which the robot can learn the association between sensory information, internal states and actions.

One contribution of AMD is that it regards attention as a type of internal state and builds a mapping between the attentional state and actions. However, AMD does not model the intrinsic mechanism of attention. The attentional selection used in AMD is directly specified by trainers.

2.6 Conclusions

This chapter has reviewed the background that is important in the field of visual attention and its robotic applications. It introduces the basic idea of the visual attention mechanism, psychological models of visual attention, computational models of visual attention and a variety of current advances in the robotic applications of visual attention.

This chapter has shown that using the visual attention mechanism to build a general and cognitive visual perception mechanism for robots is a novel and interesting topic. Several challenging issues require further research on this research topic, such as how to model the pre-attentive segmentation process for object-based attention, how to model top-down attention for object-based attention, how to model the LTM object representation used both to deduce task-relevant information for guiding top-down attention and to guide action selection, and so on. The following chapters will present how to deal with those issues in detail.

Chapter 3

Framework of the Proposed Cognitive Visual Perception

3.1 Introduction

The main objective of this thesis is to develop a cognitive visual perception paradigm for robots by using the object-based visual attention mechanism, in order that the robot can have the cognitive capability of knowing how to perceive the environment autonomously. This cognitive capability includes two aspects. The conscious aspect directs visual perception based on the task, context and knowledge learned from experience, whereas the unconscious aspect directs visual perception in the case of facing an unexpected, unusual or surprise situation. Object-based visual attention is used in this proposed perception paradigm as a core part to realize those two aspects of cognitive capability.

This chapter presents the framework of the proposed cognitive visual perception paradigm. Section 3.2 gives an overview of the proposed cognitive visual perception paradigm. Section 3.3 discusses the relationship between the proposed cognitive visual perception and active vision.

3.2 Overview of the Cognitive Visual Perception

This section presents the framework of the proposed cognitive visual perception paradigm. Figure 3.1 and Figure 3.2 illustrate a brief description and a detailed description of the framework respectively. This paradigm involves three successive stages: pre-attentive processing, attentional selection and post-attentive perception. It indicates that robotic visual perception starts from a low-level cognitive attentional selection procedure that guides attention to an object in the scene, followed by a high-level post-attentive analysis procedure that analyzes the attended object and formulates it into an internal mental representation used for further cognitive behaviors.

Pre-attentive processing stage: This stage provides the basic information used for the attentional selection stage. It consists of two successive steps. The first step is the extraction of low-level pre-attentive features. The second step is the pre-attentive segmentation that divides the scene into homogeneous proto-objects in a bottom-up, unsupervised manner. The obtained proto-objects are the fundamental units of attentional selection.

Attentional selection stage: This stage involves four modules: bottom-up competition, top-down biasing, a combination of bottom-up saliency and top-down biases, as well as estimation of proto-object based attentional activation.

The bottom-up competition module aims to model the unconscious aspect of cognitive capability. This module generates a probabilistic location-based bottom-up saliency map to simulate the bottom-up attention mechanism. This module is implemented by extending Itti's attention model [38].

The top-down biasing module aims to model the conscious aspect of cognitive capability. This module simulates the top-down attention mechanism by generating a probabilistic location-based top-down bias map based on the current task, the current context and learned knowledge. Duncan's IC hypothesis [49] is used to implement the top-down biasing module. The related aspect of the IC hypothesis used to guide top-down attention can be summarized as: by directing attention to a conspicuous cue of an

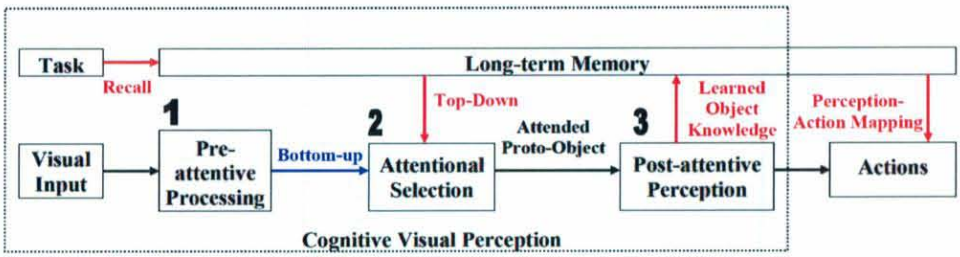


Figure 3.1: The brief framework of the proposed cognitive visual perception paradigm for robots. This paradigm involves three stages: pre-attentive processing, attentional selection and post-attentive perception. Perception and action units are connected through the selective attention mechanism.

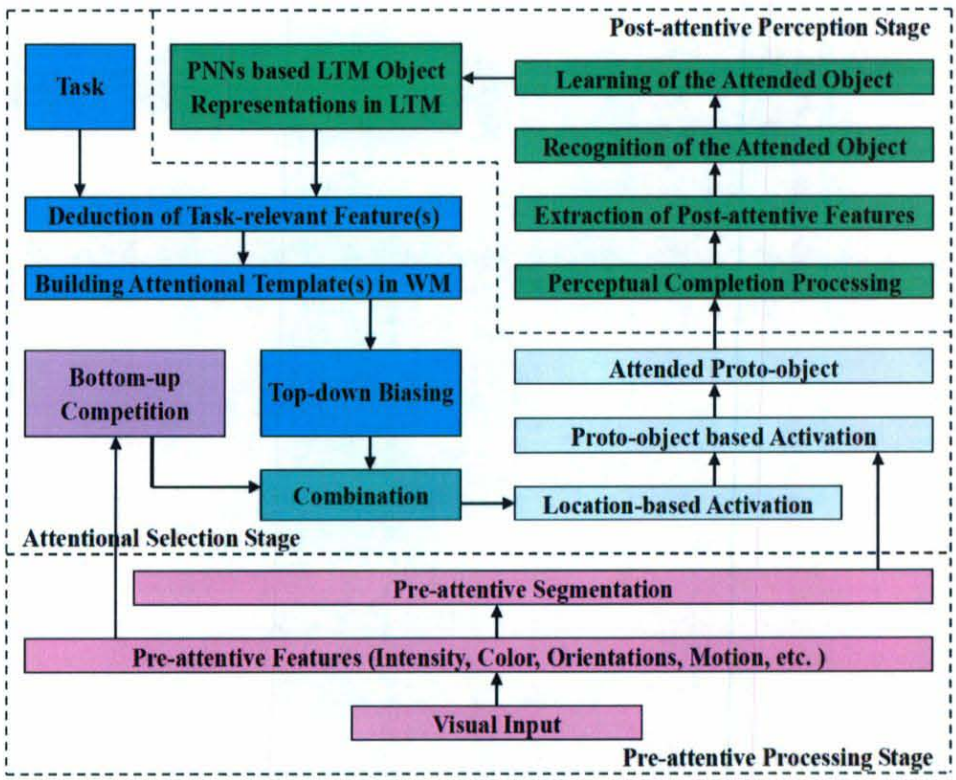


Figure 3.2: The detailed framework of the proposed cognitive visual perception paradigm for robots.

object, a competitive advantage over the whole object is produced. Accordingly, the top-down biasing module includes four steps: Deduction of a task-relevant object from the task and context, deduction of the task-relevant feature(s) from the task-relevant object, construction of the attentional template(s) and estimation of location-based top-down biases. Once a task-specific object is deduced or given directly by the task, the top-down biasing module recalls the LTM object representation from LTM, from which one or a few task-relevant feature(s) are deduced. The task-relevant feature(s) then constitute the attentional template(s). The location-based top-down biases are finally estimated by comparing attentional template(s) with corresponding pre-attentive feature(s).

Top-down biases and bottom-up saliency are combined in a probabilistic manner to yield a location-based attentional activation map.

By combining location-based attentional activation within each proto-object, a proto-object based attentional activation map is finally achieved. The most active proto-object is selected for attention.

Post-attentive perception stage: Following the attentional selection stage, the attended proto-object is sent into the post-attentive perception stage. Although the post-attentive perception stage could involve a variety of processing, this thesis asserts that the main objective of the post-attentive perception stage is to interpret the attended object in more detail. The detailed interpretation aims to produce the correct action during the process of action at the current moment and consciously guide the top-down attention during the process of perception at the next moment. Thus this stage mainly includes four modules.

The first module is perceptual completion processing. Since an object is always composed of several parts, this module is required to obtain the complete region of the attended object. It performs around the attended proto-object in the scene based on the corresponding LTM object representation.

The second module is the extraction of post-attentive features. Post-attentive features are a type of high-level features estimated based on the pre-attentive features in the

region of the attended object. In order to interpret the attended object in more detail, this module forms a high-level representation of the attended object in WM using the extracted post-attentive features.

The third module is object recognition. This module identifies what the attended object is. It can also identify which instance the attended object belongs to, which is used for further action selection during the process of action.

The fourth module is the development of LTM object representations. It is the most important module in the post-attentive perception stage. The objective of this module is to develop the corresponding LTM representation using the currently attended object. The development includes two types of functions. The first function is constructing the structure of LTM object representations and the second function is dynamically learning the corresponding LTM representation given the attended object. A robust structure of object representation is built in this thesis by using a probabilistic neural network [53]. Dynamical learning algorithms are proposed. The learned LTM object representations can be used to guide action selection at the current moment and they can also be used to guide top-down biasing, perceptual completion processing and object recognition at the next moment.

3.3 Comparison with Active Vision

It is important to note that the proposed cognitive visual perception paradigm is close to recent research on active/behavioral vision [115, 116]. Active vision situates vision within an interactive behavioral context and controls visual perception based upon the observer's activity or the present task. The similar point between the proposed paradigm and active vision is that the central aspect of both is attentional control. Thus it can be said that the proposed cognitive visual perception paradigm is a type of active vision.

However, most work on active vision does not model a general attention mechanism, but just develops a specific attentional control algorithm for each distinctive task by

predefining distinctive features, such as junctions [117], depth [118] and iconic representations [?]. Thus the attentional selection in those active vision systems is basically controlled by the programmers. That is, the robot still has no mental capability of knowing how to perceive the environment in those active vision systems.

3.4 Conclusions

This chapter introduces the framework of the proposed cognitive visual perception paradigm. It consists of three successive stages: pre-attentive processing, attentional selection and post-attentive perception. This chapter gives a brief description of each stage. Finally the relationship between the proposed cognitive visual perception paradigm and active vision is discussed.

The following chapters will present the detailed implementation of each stage in the proposed cognitive visual perception paradigm.

Chapter 4

Pre-attentive Processing

4.1 Introduction

Pre-attentive processing represents the visual processing work prior to attentional selection. Object-based attention theory [25] and a considerable body of psychological and physiological experimental evidence [45, 119] have shown that pre-attentive processing in vision mainly fulfills two functions. The first function is to extract some types of basic features, called pre-attentive features in this thesis. The second function is to carve the visual input into candidate objects, which are loose collections of the extracted pre-attentive features. The second function is called pre-attentive segmentation in this thesis.

This chapter presents the computational methods used to realize these two functions. Section 4.2 deals with the extraction of pre-attentive features, including what features can be regarded as pre-attentive features and how the computational methods are designed to extract them. Section 4.3 deals with pre-attentive segmentation, including what the properties of pre-attentive segmentation are, what psychological rules can be used to model it and how an algorithm is designed to model it. Experimental results of those two functions are also illustrated.

4.2 Extraction of Pre-attentive Features

4.2.1 Definition of Pre-attentive Features

Pre-attentive features can be defined as a type of basic properties analyzed from the input stimuli by the visual cortex prior to attentional selection. A large body of psychological research data [45, 119] has shown that there are at least four dimensions of pre-attentive features: intensity [119], colors [45], orientation energy [45] and motion [119]. Furthermore, recent psychological research has shown that at the earliest stages of visual cortical processing neurons play a role in intermediate level vision, including contour integration and surface segmentation [120]. Thus contour is also considered as a type of pre-attentive feature in this thesis. In summary, five dimensions of pre-attentive features are extracted in this proposed cognitive visual perception paradigm and they are intensity, colors, orientation energy, contour and motion.

In fact, physiological research has also provided evidence of the existence of these pre-attentive features. As discussed in section 2.2.5 in Chapter 2, V4 neurons mainly respond to colors [85], V1 neurons function as the orientation energy filters [83], MT neurons play a role in perception of motion [66], and V2 neurons respond to illusory or subjective contours [84].

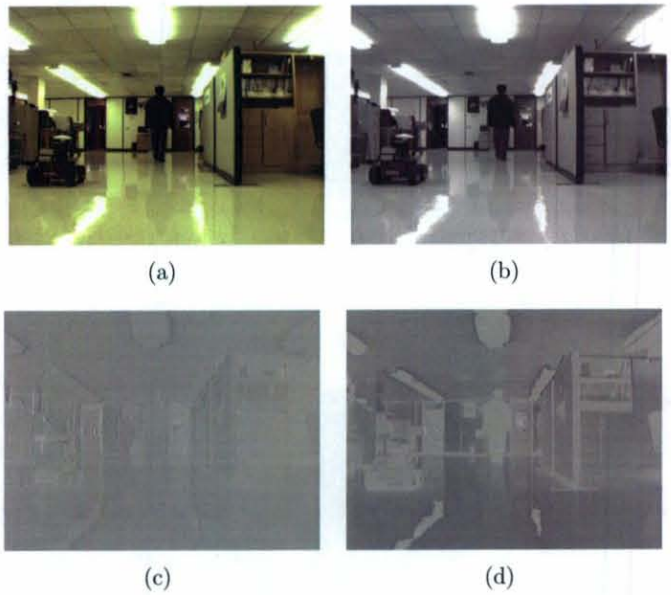
Multi-scale pre-attentive features

In order to simulate the bottom-up attention mechanism, i.e., bottom-up attentional competition in the spatial context in terms of the pre-attentive features, multi-scale pre-attentive features in each feature dimension are computed.

Note that symbol $\mathbf{F}$ is used to denote pre-attentive feature vectors in this thesis.

4.2.2 Intensity and Colors

Given the three 8-bit input color channels: red ($\mathbf{r}$), green ($\mathbf{g}$) and blue ($\mathbf{b}$), the pre-attentive feature of intensity $\mathbf{F}_{int}$ is represented by a weighted average of these three



(a) (b)

(c) (d)

Figure 4.1: Pre-attentive features in terms of intensity, red-green color pair and blue-yellow color pair on the original scale. (a) Original image. (b) Intensity. (c) Red-green pair. (d) Blue-yellow pair.

color channels:

$$\mathbf{F}_{int} = (\mathbf{r} + \mathbf{g} + \mathbf{b})/3. \tag{4.1}$$

In order that each color channel yields the maximum response for pure and fully saturated hue, four broadly-tuned color channels, including red (**R**), green (**G**), blue (**B**) and yellow (**Y**), are created:

$$\begin{aligned} \mathbf{R} &= \mathbf{r} - (\mathbf{g} + \mathbf{b})/2, \\ \mathbf{G} &= \mathbf{g} - (\mathbf{r} + \mathbf{b})/2, \\ \mathbf{B} &= \mathbf{b} - (\mathbf{r} + \mathbf{g})/2, \\ \mathbf{Y} &= (\mathbf{r} + \mathbf{g})/2 - |\mathbf{r} - \mathbf{g}|/2 - \mathbf{b}. \end{aligned} \tag{4.2}$$

The physiological findings [96] have shown that neurons in the cortex are excited by one color (e.g., red) and inhibited by another color (e.g., green). This spatial and

chromatic opponency exists for the red-green pair and blue-yellow pair in the human primary visual cortex. Therefore, as proposed in Itti's model [38], the four broadly-tuned color channels are transformed into two pre-attentive color features: red-green pair $\mathbf{F}_{rg}$ and blue-yellow pair $\mathbf{F}_{by}$. These two types of pre-attentive color features can be expressed as:

$$\mathbf{F}_{rg} = \mathbf{R} - \mathbf{G}, \tag{4.3}$$

$$\mathbf{F}_{by} = \mathbf{B} - \mathbf{Y}. \tag{4.4}$$

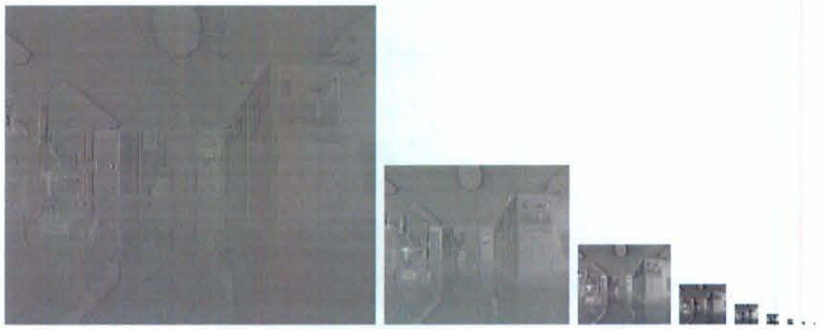
Examples of the extracted pre-attentive features in terms of intensity, red-green pair and blue-yellow pair at the original scale have been shown in Figure 4.1.

Multi-scale pre-attentive features in terms of intensity, red-green pair and blue-yellow pair are created by using the Gaussian pyramid [121], which progressively low-pass filters and subsamples those pre-attentive features. For instance, if the image size is 640×480, nine spatial scales can be created for each pre-attentive feature. The detailed implementation of the Gaussian pyramid can be seen in Appendix A. These multi-scale pre-attentive features in terms of intensity, red-green pair and blue-yellow pair can be denoted as $\mathbf{F}_{int}(l)$, $\mathbf{F}_{rg}(l)$, $\mathbf{F}_{by}(l)$ respectively, where l is the spatial scale. Examples of extracted multi-scale pre-attentive features in terms of intensity, red-green pair and blue-yellow pair are shown in Figure 4.2.

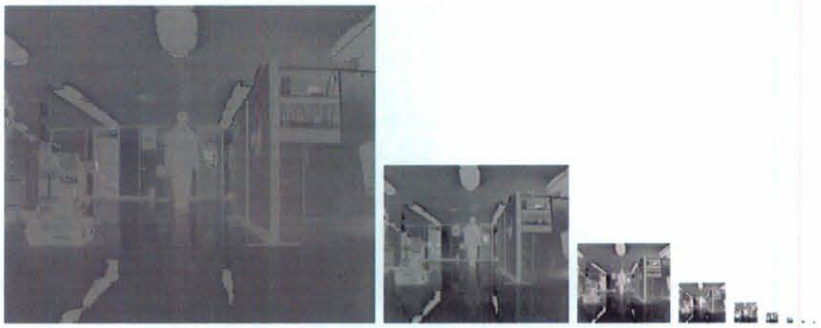
4.2.3 Orientation Energy

As discussed in section 2.2.5 in Chapter 2, the simple cells and complex cells in area V1 are responsible for extracting orientation energy in the pre-attentive stage [81]. There are two types of simple cells. One type is sensitive to lines of a particular orientation and it can be modeled by using odd-symmetric filters. The other type is sensitive to step edges of a particular orientation and it can be modeled by using even-symmetric filters. The complex cells are sensitive to the specific orientations and they can be modeled by

(a)



(b)



(c)

Figure 4.2: The multi-scale pre-attentive features in terms of intensity, red-green pair and blue-yellow pair from scale 0 to scale 8 respectively. From left to right in each row, the scale is from 0 to 8. The original image has been shown in Figure 4.1(a). (a) Multi-scale intensity features. (b) Multi-scale red-green pair features. (c) Multi-scale blue-yellow pairs features.

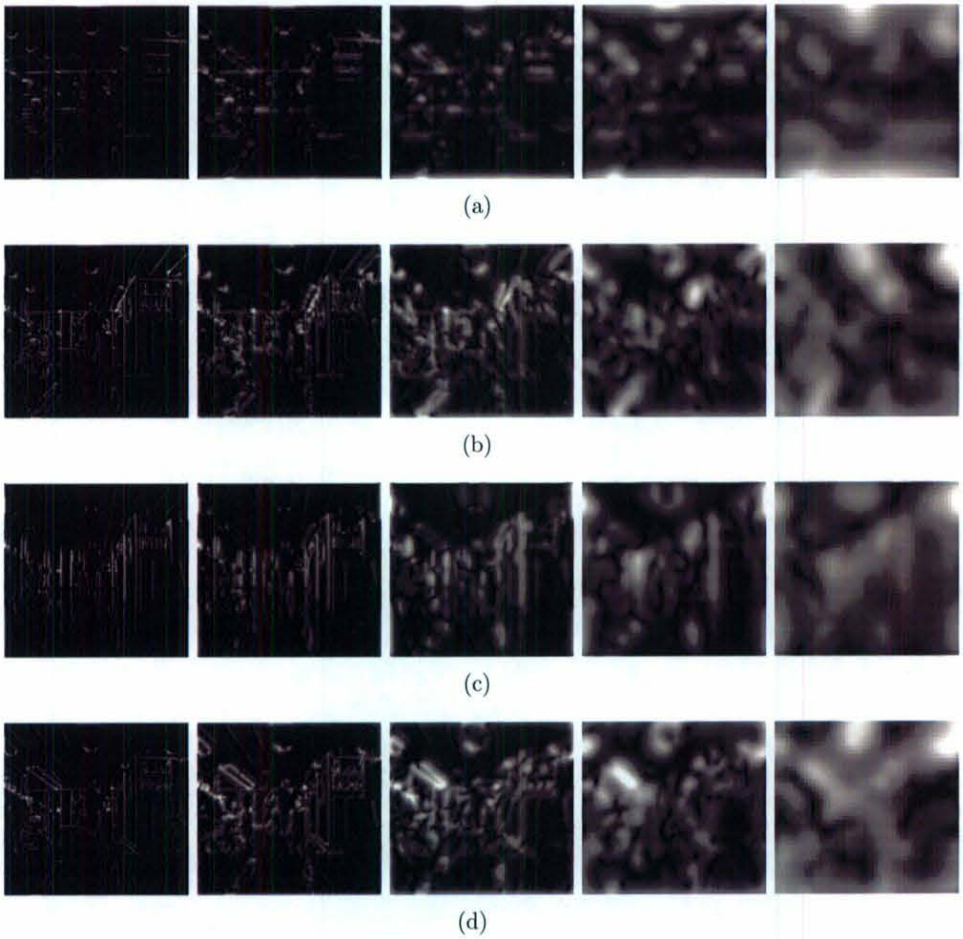


Figure 4.3: The multi-scale pre-attentive features in terms of orientation energy from scale 0 to scale 4 in four preferred orientations $\theta \in \{0°, 45°, 90°, 135°\}$. In order to clearly illustrate the features at small scales, all feature images are linearly normalized to [0,255] and are shown in the same size. From left to right in each row, the scale is from 0 to 4. The original image has been shown in Figure 4.1(a). (a) In orientation $\theta = 0°$. (b) In orientation $\theta = 45°$. (c) In orientation $\theta = 90°$. (d) In orientation $\theta = 135°$.

summing the outputs of line-sensitive simple cells and step-sensitive simple cells with the same orientations. Thus the pre-attentive feature of orientation energy in a preferred orientation can be extracted by simulating the mechanism of simple cells and complex cells.

The oriented 2-D Gabor filter, consisting of an odd-symmetric part and an even-symmetric part, can be used to approximate the sensitivity of these two types of simple

cells [122]. The detailed techniques of the 2-D Gabor filter can be seen in Appendix B.

Thus the multi-scale pre-attentive features of orientation energy, denoted as $\mathbf{F}_{o_\theta}(l)$, can be extracted by convolving the intensity image $\mathbf{F}_{int}(l)$ with a 2-D Gabor filter at scale l. According to Itti's model [38], four preferred orientations $\theta \in \{0°, 45°, 90°, 135°\}$ are used in this thesis.

Examples of multi-scale local orientation energy features are shown in Figure 4.3, where the feature images from scale 0 to scale 4 are shown.

4.2.4 Contour

The total orientation energy is used to approximate the contour feature, denoted as $\mathbf{F}_{ct}$. It is obtained by applying a pixel-wise maximum operator over all orientations:

$$F_{ct}(\mathbf{r}_i, l) = \max_{\theta \in \{0°, 45°, 90°, 135°\}} F_{o_\theta}(\mathbf{r}_i, l), \tag{4.5}$$

where $\mathbf{r}_i$ denote a pixel in the image at scale l.

4.2.5 Motion Energy

The motion processing mechanism in the visual cortex is like a derivative analyzer that estimates orientation in the space-time domain [123]. It inspires the idea to build a 3-D derivative spatio-temporal filter in order to estimate the motion energy [124], which is referred to as the spatio-temporal energy model (STEM). Correspondingly, a set of oriented Gabor spatio-temporal filters has been proposed in [125] to yield motion energy in preferred motion directions. However, the 3-D Gabor filters are computationally expensive.

Simoncelli and Anderson [126] have shown that the standard gradient-based technique can also be understood as a method for extracting motion energy. Thus a probabilistic algorithm for motion extraction is proposed in [127]. Based on the image gradient, it

estimates the conditional distribution of optical flow, which can be expressed as:

$$p(\boldsymbol{\psi}|\mathbf{d}_s, d_t) \propto \exp\{-\tfrac{1}{2}(\boldsymbol{\mu}_\psi - \boldsymbol{\psi})^T \boldsymbol{\Sigma}_\psi^{-1}(\boldsymbol{\mu}_\psi - \boldsymbol{\psi})\} \ , \tag{4.6}$$

where $\boldsymbol{\psi}$ is the optical flow vector, $p(\boldsymbol{\psi}|\mathbf{d}_s, d_t)$ is the conditional distribution of the optical flow given the spatial and temporal derivatives, and

$$\boldsymbol{\Sigma}_\psi = [\mathbf{d}_s(\mathbf{d}_s^T \boldsymbol{\Sigma}_s \mathbf{d}_s + \boldsymbol{\Sigma}_t)^{-1}\mathbf{d}_s^T + \boldsymbol{\Sigma}_p^{-1}]^{-1}, \tag{4.7}$$

$$\boldsymbol{\mu}_\psi = -\boldsymbol{\Sigma}_\psi \mathbf{d}_s(\mathbf{d}_s^T \boldsymbol{\Sigma}_s \mathbf{d}_s + \boldsymbol{\Sigma}_t)^{-1} d_t, \tag{4.8}$$

$$\mathbf{d}_s = \begin{pmatrix} d_x & d_y \end{pmatrix}^T, \tag{4.9}$$

where d_x, d_y and d_t are measures of spatial and temporal derivatives respectively, $\boldsymbol{\mu}_\psi$ and $\boldsymbol{\Sigma}_\psi$ are mean and covariance respectively of the conditional distribution of the optical flow, $\boldsymbol{\Sigma}_s$, $\boldsymbol{\Sigma}_t$ and $\boldsymbol{\Sigma}_p$ are covariances of measure uncertainty of spatial derivatives, measure uncertainty of temporal derivative and prior uncertainty respectively.

A first-order derivative operator is used to calculate derivatives d_x, d_y and d_t:

$$\mathbf{D} = \begin{pmatrix} 1 & -1 \end{pmatrix}. \tag{4.10}$$

The pre-attentive feature in terms of motion energy, denoted as $\mathbf{F}_{mv}$, can be estimated by using the norm of the mean vector of $p(\boldsymbol{\psi}|\mathbf{d}_s, d_t)$:

$$F_{mv}(\mathbf{r}_i) = \|\boldsymbol{\mu}_\psi(\mathbf{r}_i)\|, \tag{4.11}$$

where $\|\cdot\|$ is the Euclidean norm operator.

The above procedures are performed at each scale to yield multi-scale pre-attentive features in terms of motion energy, denoted as $\mathbf{F}_{mv}(l)$.

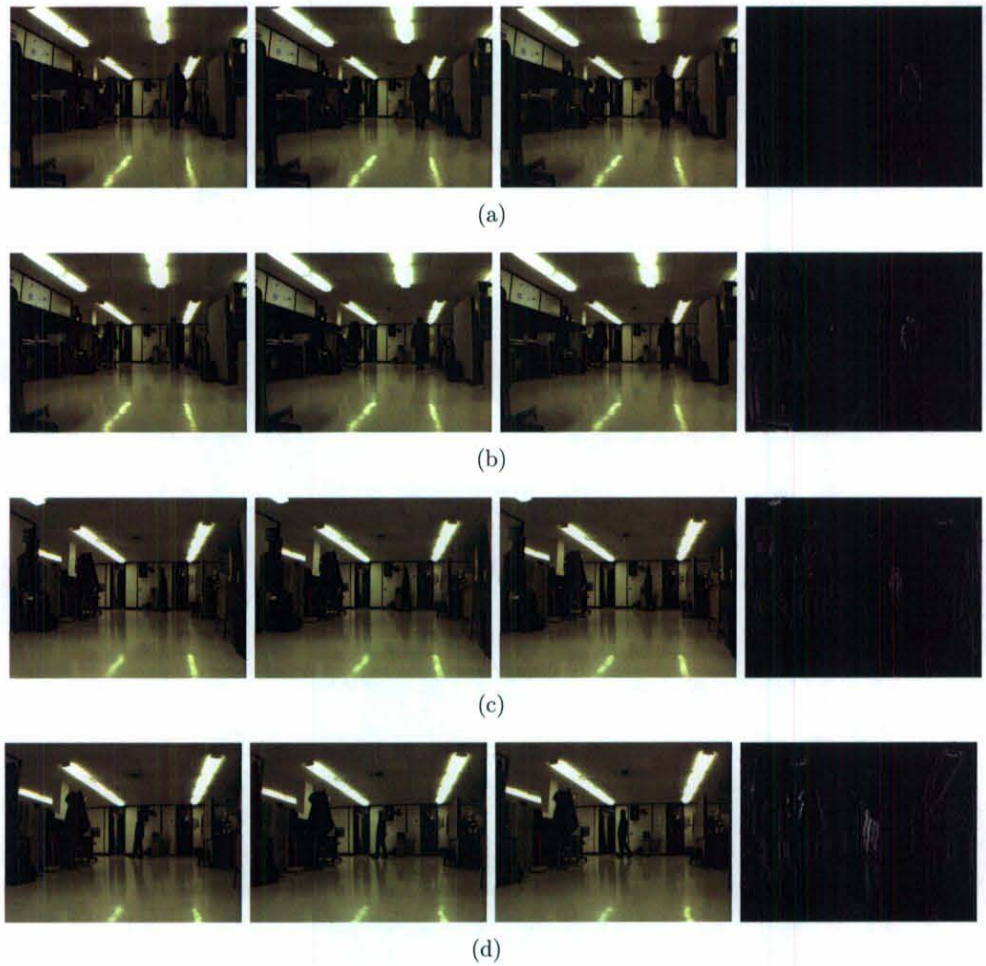


Figure 4.4: Pre-attentive features in terms of motion energy. The brightness represents the motion energy. There are two kinds of motion, including foreground and background, since the camera is also moving. The first three columns in each row are three successive original images. The last column in each row is the motion energy image.

Figure 4.4 shows some examples of pre-attentive features in terms of motion energy extracted from successive natural images at the original scale. This figure shows the effectiveness of the proposed extraction method in a set of different motion patterns (i.e., speed and direction) of foreground and background.

4.3 Pre-attentive Segmentation

4.3.1 Definition of Pre-attentive Segmentation

According to the object-based attention theory [25,34,45,128], *pre-attentive segmentation* can be defined as an unsupervised perceptual grouping process with a certain degree of accuracy prior to the attentional selection. It results in some groupings, which are the units of attentional selection. This definition indicates three properties of pre-attentive segmentation: existence, primitiveness and automaticity. That is, it is certain that the visual scene is divided into groupings in the pre-attentive stage in an automatic way whereas those groupings are primitive. This thesis calls those primitive groupings proto-objects.

Thus there exist two issues for modeling pre-attentive segmentation. The first one is how to define proto-objects and the second one is how to design an effective algorithm to implement the pre-attentive segmentation.

In the next subsections, the psychological rules used to guide perceptual grouping are firstly introduced. Based on those psychological rules and neurobiological properties of pre-attentive segmentation, proto-objects are then defined. After reviewing the existing techniques of unsupervised image segmentation, a novel algorithm for pre-attentive segmentation is finally proposed.

4.3.2 Gestalt Principle

The Gestalt principle [58] proposed in psychology is widely used to understand perceptual grouping and to guide the design of perceptual grouping algorithms. It is important to note that perceptual grouping includes both unsupervised and supervised manners. The Gestalt principle mainly includes the following rules:

- **Proximity:** The pixels which are closer in the space domain are grouped strongly together. This rule is always used for unsupervised perceptual grouping.

- **Similarity:** The pixels or blocks of pixels which are similar in terms of at least one attribute tend to be grouped together. The attributes can be color, size, orientation, direction or speed of motion and so on. Thus similarity is always regarded as a general rule for unsupervised perceptual grouping.

- **Continuity:** This rule works for lines or edges. All else being equal, elements that can be seen as smooth continuations of each other tend to be grouped together. Features, such as line terminations and line intersections, are required to realize this rule. This rule is always used for unsupervised perceptual grouping.

- **Closure:** All else being equal, elements forming a closed figure tend to be grouped together. Closure feature is required for this rule. This rule is always used for unsupervised perceptual grouping.

- **Past experience:** If elements have been previously associated with each other in prior views, they will tend to be seen as grouped in the present situation. The learned knowledge stored in LTM is required for this rule. Unlike the above rules that guide the unsupervised grouping, this rule guides the perceptual grouping in a supervised way.

It is important to note that which rules of the Gestalt principle can be used for perceptual grouping is partly dependent on which features are available.

4.3.3 Definition of Proto-objects

It is obvious that pre-attentive segmentation can only use pre-attentive features, including intensity, colors, orientation energy, contour and motion energy. Thus proximity and similarity rules can be used for pre-attentive segmentation. It can be seen that those two rules are able to divide the scene into homogeneous regions. Thus the *proto-objects* obtained by pre-attentive segmentation can be defined as the homogeneous regions obtained by using proximity and similarity rules.

4.3.4 Background of Unsupervised Segmentation Algorithms

Designing an algorithm for unsupervised image segmentation is a challenging issue. Current computational methods for unsupervised image segmentation can be categorized as boundary-based and region-based.

Boundary-based approaches, such as [3], are based on the fact that edges belonging to a single object are in adjacent positions. This is consistent with the proximity rule. However, methods in this category are not robust in the cluttered environment.

Region-based approaches are dependent on the similarity rules and widely used for unsupervised image segmentation. There are mainly three types of algorithms in this category: global optimization based, region growing and region merging.

The objective of global optimization based algorithms, such as [129–134], is to find a grouping solution by globally optimizing a criterion. A good example of the algorithms of this type is the normalized cut approach [132–134]. It divides the image into regions by optimizing the normalized cut criterion that measures the similarity between the different groups as well as the similarity within the groups. Several similarity measurement criteria, such as color, orientation and texture, have been proposed for this algorithm. However, the disadvantage of these global optimization based algorithms is that they are computationally expensive due to the computation of high-dimensional matrices during the optimization procedure.

Region growing algorithms, such as [135, 136], firstly select several interesting pixels or regions as seeds, then search for similar neighbors around each seed gradually, and finally organize each seed and its neighbors into an identical group. The challenging issue of algorithms of this type is how to select the seeds, since these seeds are the initial condition for segmentation. The selection of seeds is the most important factor to decide the segmentation performance.

In contrast to the region growing algorithms, region merging algorithms, such as [54, 55, 137–140], do not select seeds, but consider each pixel or regular region equally. The algorithms of this type hierarchically merge the similar pixels or regions into the

same group. A good example of the region merging algorithms is pyramid based segmentation [54, 55, 138–140]. This algorithm hierarchically builds each level of the pyramid by accumulating similar local nodes at the level below, with the result that the final global segments emerge in this process as they are represented by single nodes at some levels. One advantage of this pyramid based algorithms is that it is computationally fast. The other advantage is that the accumulation process of the pyramid can be used to simulate the synchronization mechanism in the human visual system. Synchronization is an effective signal for perceptual grouping. The firing activity of the scattered neurons, which code different features of one segment, is synchronized in a way that is appropriate to the prevailing context [141]. Analogous to the synchronization mechanism, nodes at each level of the pyramid in the pyramid based segmentation algorithm can be seen as the receptive fields of neurons with the corresponding spatial size. The accumulation process, in which the similarity is estimated by integrating different features, can be seen as the synchronization process in the spatial context.

Thus this thesis proposes a pre-attentive segmentation algorithm by extending the hierarchical pyramid based segmentation technique.

4.3.5 Proposed Pre-attentive Segmentation Algorithm

Why Use An Irregular Pyramid?

Pyramidal structures, including regular and irregular pyramids, have been used in the algorithms of image segmentation [54, 55, 138–140, 142]. The construction of a regular pyramid is strongly constrained by a geometric criterion in that the neighbor nodes at the same scale are always merged in spite of the similarity between them. The result is that the segmentation methods based on the regular pyramid technique [138] have difficulties in the case that the image has an irregular structure. In contrast, the construction of an irregular pyramid is constrained not only by the geometric criterion but also by the similarity. Thus this thesis employs the irregular pyramid technique to design

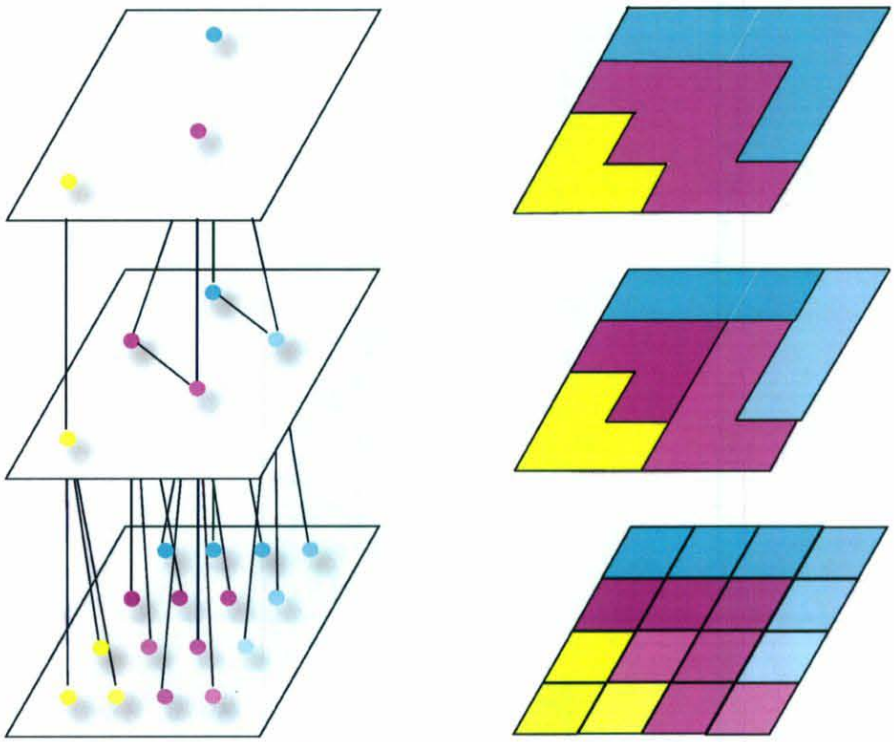


Figure 4.5: A brief graphic description of image segmentation using the irregular pyramid technique. The aggregation process of the irregular pyramid is shown from bottom to top. In the left figure, the aggregation process is represented by vertices and each circle represents a vertex. In the right figure, the aggregation process is represented by image pixels and each block represents an image pixel. It can be seen that the image is partitioned into three irregular regions once the aggregation process is finished. The color of each vertex and block represents the feature value. Note that each son vertex has only one parent vertex in this illustrative figure in order to show the aggregation process in a simple way. In fact, each son vertex can have at least one parent vertex in the proposed pre-attentive segmentation algorithm. The detailed techniques are discussed in the following text.

the pre-attentive segmentation algorithm. Figure 4.5 show a simple example of image segmentation using the irregular pyramid based technique.

Representation of the Irregular Pyramid

The graph technique is used to represent each level of the irregular pyramid. Level l of the irregular pyramid is represented by a graph $\mathbf{G}_l = (\mathbf{V}_l, \mathbf{E}_l)$, consisting of vertices $v \in \mathbf{V}_l$ and intra-level edges $e^l \in \mathbf{E}_l$. Intra-level edges represent the similarity between a vertex and its neighbors at the same level. The symbol $\mathbb{N}_i$ is used to denote the neighbor set of a vertex v_i.

Note that the index (e.g., i) for a vertex (e.g., v_i) is determined at the base level $l = 0$ and it remains unchanged at high levels if the vertex survives at these high levels. An example of this case can be seen in Figure 4.7(a).

Each vertex in $\mathbf{G}_l$ is also linked to its parent vertices in $\mathbf{G}_{l+1}$ by inter-level edges $p^{[l,l+1]}$, which represent the membership of a son vertex in $\mathbf{G}_l$ to its parent vertices in $\mathbf{G}_{l+1}$. A graphic illustration of inter-level edges and intra-level edges of the irregular pyramid is shown in Figure 4.6.

Strength of Intra-level Edges

Estimating the strength of intra-level edges, i.e., the similarity between a vertex and its neighbors at the same level, is very important for the final performance of pre-attentive segmentation.

The first issue related to the similarity measures is which features are used. Since this measurement is used for the pre-attentive segmentation, only pre-attentive features can be used. As intensity and colors can be easily used to estimate the similarity for region-based segmentation approaches, the proposed pre-attentive segmentation algorithm employs the pre-attentive features of intensity, red-green pair and blue-yellow pair. An aggregate

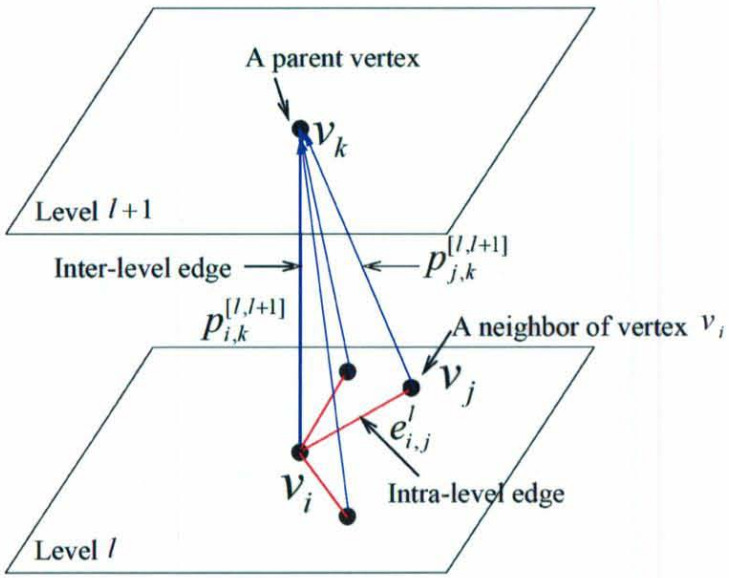


Figure 4.6: A graphic description of inter-level edges and intra-level edges of the irregular pyramid. Note that each son vertex has only one parent vertex in this illustrative figure in order to show them in a simple way. In fact, each son vertex can have at least one parent vertex in the proposed pre-attentive segmentation algorithm.

feature vector of a vertex v_i, denoted as $\hat{\mathbf{F}}_i^l$, can be built as:

$$\hat{\mathbf{F}}_i^l = \begin{pmatrix} \hat{F}_{int,i}^l \\ \hat{F}_{rg,i}^l \\ \hat{F}_{by,i}^l \end{pmatrix}, \tag{4.12}$$

where $\hat{F}_{int,i}^l$, $\hat{F}_{rg,i}^l$ and $\hat{F}_{by,i}^l$ represent the aggregate features of a vertex v_i at a pyramidal level l in terms of intensity, red-green pair, blue-yellow pair respectively.

The second issue is which type of similarity measure should be used. A *similarity measure* can be generally defined as the distance between two stimuli. A number of similarity measures have been proposed and they can be categorized into two groups: deterministic and probabilistic. Deterministic similarity measures, such as Euclidean distance, aim to estimate the distance between two deterministic stimuli. In contrast, probabilistic

similarity measures aim to estimate the distance between two stochastic stimuli. There are mainly four types of probabilistic similarity measures: Mahalanobis distance [143], Jeffreys divergence [144], Kullback-Leibler (KL) divergence [145] and Bhattacharyya distance [146].

Which type of similarity measure is suitable for estimating the strength of the intra-level edge is dependent on the properties of the two vertices linked by the edge.

At the base level of the irregular pyramid, each vertex is a single point of the image, rather than an aggregation of vertices. Therefore, the features of each vertex at the base level can be seen as deterministic and exponential Euclidean distance is used to estimate the strength of intra-level edges at the base level $l = 0$:

$$e^0_{i,j} = \exp(-||\hat{\mathbf{F}}^0_i - \hat{\mathbf{F}}^0_j||), \tag{4.13}$$

where v_j is a neighbor of vertex v_i at the base level, $e^0_{i,j}$ denotes the strength of the intra-level edge between vertices v_i and v_j at the base level $l = 0$, and $||\cdot||$ is the multi-dimensional Euclidean distance operator.

In contrast, at higher levels of the irregular pyramid, each vertex represents an aggregation of vertices at the lower levels. Thus the features of each vertex at higher levels can be seen as probabilistic distributions. This indicates that a probabilistic similarity measure should be used. There is an important requirement of this probabilistic similarity measure. It should have the ability to measure the probabilistic distance between two probabilistic distributions since the features of two vertices are both probabilistic. Based on the facts that Mahalanobis distance cannot measure the similarity between two probabilistic distributions, and that KL divergence is an asymmetric measure, Jeffreys divergence and Bhattacharyya distance can be used as candidates. This thesis selects Bhattacharyya distance for measuring the strength of intra-level edges, since it computationally costs less and is easier to implement than Jeffreys divergence.

Bhattacharyya distance, denoted as D_B, between two Gaussian distributions can be

expressed as:

$$D_B = \frac{1}{8}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T \boldsymbol{\Sigma}_{1,2}^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2) + \frac{1}{2}\ln\left(\frac{\det(\boldsymbol{\Sigma}_{1,2})}{\sqrt{\det(\boldsymbol{\Sigma}_1)\det(\boldsymbol{\Sigma}_2)}}\right), \tag{4.14}$$

where $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$ are mean vectors of those two Gaussian distributions, $\boldsymbol{\Sigma}_1$ and $\boldsymbol{\Sigma}_2$ are covariance matrices of those two Gaussian distributions, and

$$\boldsymbol{\Sigma}_{1,2} = \frac{\boldsymbol{\Sigma}_1 + \boldsymbol{\Sigma}_2}{2}. \tag{4.15}$$

Assuming that the feature's probabilistic distribution of a vertex at higher levels in the irregular pyramid is Gaussian, the strength of intra-level edges at higher levels $l > 0$ can be therefore estimated as:

$$e_{i,j}^l = \exp\left\{ -\frac{1}{8}(\hat{\mathbf{F}}_i^l - \hat{\mathbf{F}}_j^l)^T(\hat{\boldsymbol{\Sigma}}_{i,j}^l)^{-1}(\hat{\mathbf{F}}_i^l - \hat{\mathbf{F}}_j^l) - \frac{1}{2}\ln\left(\frac{\det(\hat{\boldsymbol{\Sigma}}_{i,j}^l)}{\sqrt{\det(\hat{\boldsymbol{\Sigma}}_i^l)\det(\hat{\boldsymbol{\Sigma}}_j^l)}}\right)\right\}, \tag{4.16}$$

where $e_{i,j}^l$ denotes the strength of an intra-level edge between vertices v_i and v_j at level l, $\hat{\boldsymbol{\Sigma}}_i^l$ and $\hat{\boldsymbol{\Sigma}}_j^l$ are covariance matrices of aggregate features of vertex v_i and vertex v_j respectively at level l, and

$$\hat{\boldsymbol{\Sigma}}_{i,j}^l = \frac{\hat{\boldsymbol{\Sigma}}_i^l + \hat{\boldsymbol{\Sigma}}_j^l}{2}. \tag{4.17}$$

Scale-invariance of the Similarity Measure

One advantage of this probabilistic similarity measure is its scale-invariance during the pyramidal aggregation procedure. With the accumulation of the son vertices to the parent vertex, the feature data (e.g., mean and covariance) of the parent vertex are changing, i.e., the scale of the measurement space is changing. Since Bhattacharyya distance takes into account the correlations (i.e., covariances) of the data sets, the estimated similarity is approximately scale-invariant during the pyramidal aggregation procedure.

Initialization of the Base Level

The base level $l = 0$ of the irregular pyramid is initialized by using pre-attentive features at working scale $l_{wk} = 2$ (see details about the working scale in section 5.2.3). An 8-connected graph is used in this algorithm.

The Aggregation Process

The aggregation process of the irregular pyramid aims to hierarchically build each level of the pyramid by accumulating similar neighbor nodes at the level below. During this process, the final global segments emerge as they are represented by single nodes at some level. This process consists of four procedures: decimation, estimating the strength of inter-level edges, estimating the aggregate features and searching for neighbors. The detailed implementation of these four procedures are given in the following paragraphs.

Procedure 1: Decimation

The first procedure is decimation, in which a subset of $\mathbf{V}_l$ (i.e., a set of surviving vertices) is selected from the graph $\mathbf{G}_l$ to build the graph $\mathbf{G}_{l+1}$. The objective of the decimation procedure is that the accumulated parent level $\mathbf{G}_{l+1}$ can represent the son level $\mathbf{G}_l$ as enough as possible.

Two rules have been proposed to constrain the decimation procedure in [142]:

- *Rule 1:* The decimation must be maximal: Any two neighbor vertices cannot both survive to the next high level. This can be mathematically expressed as: $\forall v_i \in \mathbf{V}_l$ and $\forall v_j \in \mathbb{N}_i$, if $v_i \in \mathbf{V}_{l+1}, \{v_j \in \mathbf{V}_{l+1}\} = \emptyset$.

- *Rule 2:* Any son vertex must have at least one parent vertex at the next high level. This can be mathematically expressed as: $\forall v_i \in \mathbf{V}_l$, $\{v_i \in \mathbf{V}_{l+1}\} \cup \{\mathbb{N}_i \cap \mathbf{V}_{l+1}\} \neq \emptyset$.

Those two rules can be illustrated in Figure 4.7.

Based on these two rules, a stochastic pyramid decimation (SPD) algorithm [140,142] has been proposed. In the SPD algorithm, a random value is first associated with each

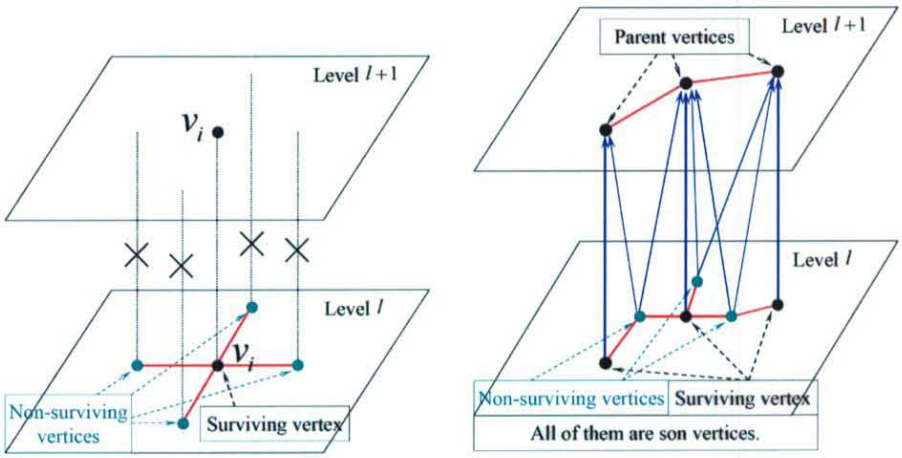


Figure 4.7: A graphic description of two rules used in the decimation procedure during the pyramidal aggregation. (a) Illustration of *Rule 1*: any two neighbor vertices cannot both survive to the next high level. In this sub-figure, v_i is a surviving vertex and all of its four neighbor vertices at level l do not survive. (b) Illustration of *Rule 2*: any son vertex must have at least one parent vertex at the next high level. This sub-figure shows that each son vertex at level l has at least one parent vertex at level $l+1$. Blue solid lines represent the inter-level edges and the widths of those lines represent the strength of the inter-level edges. In both sub-figures, red solid lines represent the intra-level edges.

vertex. Based on these random values, the selection of survivors are iteratively performed with the constraints of *rule 1* and *rule 2*.

By extending the SPD algorithm, this thesis proposes a new decimation procedure, called data-driven decimation (DDD). In the DDD algorithm, the similarity between a vertex and its neighbors is used to determine the selection of survivors instead of the random value used in the SPD algorithm.

Two logical variables are associated with each vertex v_i at level l. Variable a_i^{l+1} indicates if the vertex v_i will survive at level $l+1$. Variable b_i^{l+1} indicates if *rule 2* is not satisfied for the vertex v_i. This DDD procedure consists of two routines.

The first routine is called *selection of local maxima*. This routine is not recursive, i.e., it only works in the first iteration of the decimation procedure. According to *rule 1*, some surviving vertices can be first selected if they are local maxima. Therefore, vertex

$v_i \in \mathbf{V}_l$ will survive at level $l+1$ if it has the maximum similarity among its neighbors. This routine is implemented by labeling those two logical variables. This routine can be mathematically expressed as:

$$\begin{cases} a_i^{l+1,n=1} = true, & \text{if } \tilde{e}_i^l > \tilde{e}_j^l, \forall v_j \in \mathbb{N}_i \\ a_i^{l+1,n=1} = false, & \text{otherwise} \end{cases}, \qquad (4.18)$$
$$b_i^{l+1,n=1} = \overline{a}_i^{l+1,n=1} \wedge_{v_j \in \mathbb{N}_i} \overline{a}_j^{l+1,n=1},$$

where n denotes the index of the decimation iteration, $\tilde{e}_i^l$ is the sum of strength of intra-level edges of the vertex v_i and it can be mathematically expressed as: $\tilde{e}_i^l = \sum_j e_{i,j}^l$ where $v_j \in \mathbb{N}_i$, $\tilde{e}_j^l$ is the sum of strength of intra-level edges of the vertex v_j and it can be mathematically expressed as: $\tilde{e}_j^l = \sum_{j'} e_{j,j'}^l$ where $v_{j'} \in \mathbb{N}_j$, $\wedge$ denotes the "logic and" operator, and $\overline{a}$ denotes the negative of a.

Since the surviving vertices selected by the first routine are sparse in most cases, another routine is further required to satisfy *rule 2* and this routine is called *selection of local sub-maxima*. During this routine, some vertices are iteratively selected as the rest of survivors with the constraints of *rule 1* and *rule 2*. This routine can be mathematically expressed as:

$$\begin{cases} a_i^{l+1,n} = a_i^{l+1,n-1} \vee b_i^{l+1,n-1}, & \text{if } \tilde{e}_i^l > \tilde{e}_j^l, \forall v_j \in \mathbb{N}_i \\ a_i^{l+1,n} = a_i^{l+1,n-1}, & \text{otherwise} \end{cases}, \qquad (4.19)$$
$$b_i^{l+1,n} = \overline{a}_i^{l+1,n} \wedge_{v_j \in \mathbb{N}_i} \overline{a}_j^{l+1,n},$$

where n denotes the index of the decimation iteration and $n > 1$, $\tilde{e}_i^l = \sum_j e_{i,j}^l$ where $v_j \in \{\mathbb{N}_i \wedge b_j^{l+1,n-1}\}$, $\tilde{e}_j^l = \sum_{j'} e_{j,j'}^l$ where $v_{j'} \in \{\mathbb{N}_j \wedge b_{j'}^{l+1,n-1}\}$, and $\vee$ denotes the "logic or" operator.

The second routine is iterated until $\forall v_i, b_i^{l+1,n}$ are false. Convergence of this routine is guaranteed since each iteration strictly increases the number of surviving vertices.

Procedure 2: Estimating the Strength of Inter-level Edges

The second procedure is to estimate the strength of inter-level edges $p^{[l,l+1]}$ by satisfying the following conditions [55]:

1. $\sum_k p_{i,k}^{[l,l+1]} = 1$, $\forall v_k \in \mathbf{V}_{l+1}$, for every $v_i \in \mathbf{V}_l$.
2. $p_{i,k}^{[l,l+1]}$ is proportional to $e_{i,k}^{l}$, $\forall v_k \in \mathbf{V}_{l+1}$, for every $v_i \in \mathbf{V}_l \backslash \mathbf{V}_{l+1}$.
3. $p_{i,i}^{[l,l+1]} = 1$, for $v_i \in \mathbf{V}_l \bigcap \mathbf{V}_{l+1}$.

Procedure 3: Estimating the Aggregate Features

The third procedure is to estimate the aggregate features and covariances of vertices $v_k \in \mathbf{V}_{l+1}$ based on the strength of inter-level edges. The aggregate features are first estimated as:

$$\hat{\mathbf{F}}_k^{l+1} = \sum_i p_{i,k}^{[l,l+1]} \hat{\mathbf{F}}_i^l, \tag{4.20}$$

where $\hat{\mathbf{F}}_k^{l+1}$ denotes the aggregate feature vector of a vertex $v_k \in \mathbf{V}_{l+1}$, and $\hat{\mathbf{F}}_i^l$ denotes the aggregate feature vector of a vertex $v_i \in \mathbf{V}_l$.

Assuming that the pre-attentive features are independent, the covariance matrix of aggregate features of a vertex is diagonal. The diagonal entries $\hat{\sigma}_{k,f}^{l+1}$ of the covariance matrix $\hat{\mathbf{\Sigma}}_k^{l+1}$ of a vertex $v_k \in \mathbf{V}_{l+1}$ are then estimated as:

$$\hat{\sigma}_{k,f}^{l+1} = \sqrt{\sum_i p_{i,k}^{[l,l+1]} \left[(\hat{F}_{i,f}^{l})^2 + (\hat{\sigma}_{i,f}^{l})^2 \right] - (\hat{F}_{k,f}^{l+1})^2}, \tag{4.21}$$

where $f \in \{int, rg, by\}$ and $\hat{\sigma}_{i,f}^{0} = 0, \forall i, \forall f$.

The non-diagonal entries of the covariance matrix $\hat{\mathbf{\Sigma}}_k^{l+1}$ of each vertex v_k at level $l+1$ are set to 0.

Procedure 4: Search for Neighbors

The fourth procedure is to search for neighbors of each vertex $v_k \in \mathbf{V}_{l+1}$ and simultaneously estimate the strength of intra-level edges $e_{k,k'}^{l+1} \in \mathbf{E}_{l+1}$ at level $l+1$ using (4.16).

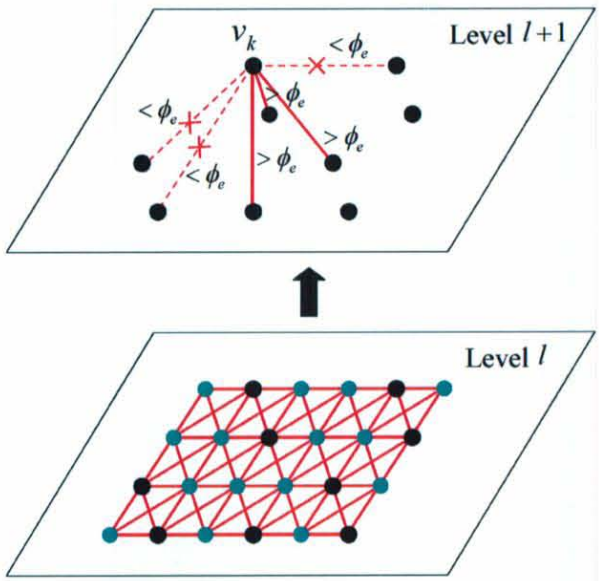


Figure 4.8: The similarity-driven neighbor search procedure in the pre-attentive segmentation algorithm. Vertex v_k is used as an example to illustrate the neighbor search procedure at level $l+1$. Red lines, including solid and dash lines, represent the candidate intra-level edges of v_k. Finally, the red solid lines are selected as the intra-level edges of v_k since their strength is larger than the threshold. At level l, the black circles represent the vertices that survive at level $l+1$ and the green circles represent the vertices that do not survive at level $l+1$. At level $l+1$, the black circles represent the vertices at that level.

A new neighbor search method is proposed in this thesis and it is graphically illustrated in Figure 4.8. This method not only uses the graphic constraint for neighbor search, but also considers the similarity constraint in the sense that the candidate neighbors should be similar enough to the center vertex. This method is called a *similarity-driven neighbor search procedure* and it can be expressed as: a vertex $v_{k'} \in \mathbf{V}_{l+1}$ is selected as a neighbor of vertex $v_k \in \mathbf{V}_{l+1}$ if $e^{l+1}_{k,k'} > \phi_e$. Parameter ϕ_e is a similarity threshold, which controls the precision of pre-attentive segmentation. Since the proposed pre-attentive segmentation algorithm uses Bhattacharyya distance as a similarity measure, which is approximately scale-invariant, the parameter ϕ_e can be a fixed value for all pyramidal levels. This thesis empirically sets this parameter as $\phi_e = 1.5$.

It is important to note that v_k and $v_{k'}$ must have a connection path at level l and

path-length$(k, k') = \{2, 3\}$ is adopted to search for neighbors in this thesis.

Emergence of Proto-objects

In the case that no neighbors are found for a vertex $v_k \in \mathbf{V}_{l+1}$, it is labeled as a new proto-object if its area is larger than the predefined threshold ϕ_a; Otherwise, it is merged into the closest and most similar vertex at the same level. A proto-object is denoted as $\mathbf{R}_g$, where g is the index of the proto-object. This thesis sets the area threshold $\phi_a = (A_x A_y/1000)$, where (A_x, A_y) are image width and height respectively at the base level. The area of a vertex at a level can be iteratively calculated as follows:

$$A_k^{l+1} = \sum_i p_{i,k}^{[l,l+1]} A_i^l \tag{4.22}$$

where A_k^{l+1} is the area of vertex $v_k \in \mathbf{V}_{l+1}$, and A_i^l is the area of vertex $v_i \in \mathbf{V}_l$ which is initialized by 1 at the base level $l = 0$.

Final Segmentation

The construction of the full pyramid is finished once all vertices at a level have no neighbors and the level is denoted as l_{top}. Finally, the membership of each vertex at the base level to each proto-object is iteratively calculated from l_{top} to the base level $l = 0$. A membership vector $[u_{i,1}^l, ..., u_{i,g}^l, ...]$ is assigned to a vertex v_i at level l in order to denote its membership to each proto-object. Each entry of the state vector of a vertex $v_k \in \mathbf{V}_{l_{top}}$ is initialized at level l_{top} as:

$$u_{k,g}^{l_{top}} = \begin{cases} 1 & \text{if } v_k \in \mathbf{R}_g \\ 0 & \text{otherwise} \end{cases}. \tag{4.23}$$

The membership vector of a vertex v_i at the base level is achieved by iteratively using

(4.24):

$$u_{i,g}^{l-1} = \sum_{k} p_{i,k}^{[l-1,l]} u_{k,g}^{l}, \quad \forall g. \tag{4.24}$$

Each vertex v_i at the base level is finally assigned to a proto-object according to its membership vector $[u_{i,1}^{l=0}, ..., u_{i,g}^{l=0}, ...]$.

Results of the Proposed Pre-attentive Segmentation Algorithm

The proposed pre-attentive segmentation algorithm was tested using natural images obtained in different scenes and under different settings, such as changing lighting conditions. Some example results are shown in Figure 4.9. The discussion of these results is given in section 4.3.7 in this chapter.

4.3.6 Principal Axes of Proto-objects

Once proto-objects are obtained, their principal axes are calculated as:

$$\theta_{\mathbf{R}_g} = \frac{1}{2}\text{atan2}\left(\frac{2\overline{M}_{11}}{\overline{M}_{20} - \overline{M}_{02}}\right) + 90°, \tag{4.25}$$

where $\theta_{\mathbf{R}_g}$ denotes the direction of the principal axis of the proto-object $\mathbf{R}_g$ with respect to Y-axis in the image coordinate system, and $\overline{M}_{11}$, $\overline{M}_{20}$ and $\overline{M}_{02}$ are 2-order moments relative to the center of mass of that proto-object.

Those moments can be calculated respectively as:

$$\overline{M}_{11} = M_{11} - \frac{M_{10}M_{01}}{M_{00}}, \tag{4.26}$$

$$\overline{M}_{02} = M_{02} - \frac{M_{01}^2}{M_{00}}, \tag{4.27}$$

$$\overline{M}_{20} = M_{20} - \frac{M_{10}^2}{M_{00}}, \tag{4.28}$$

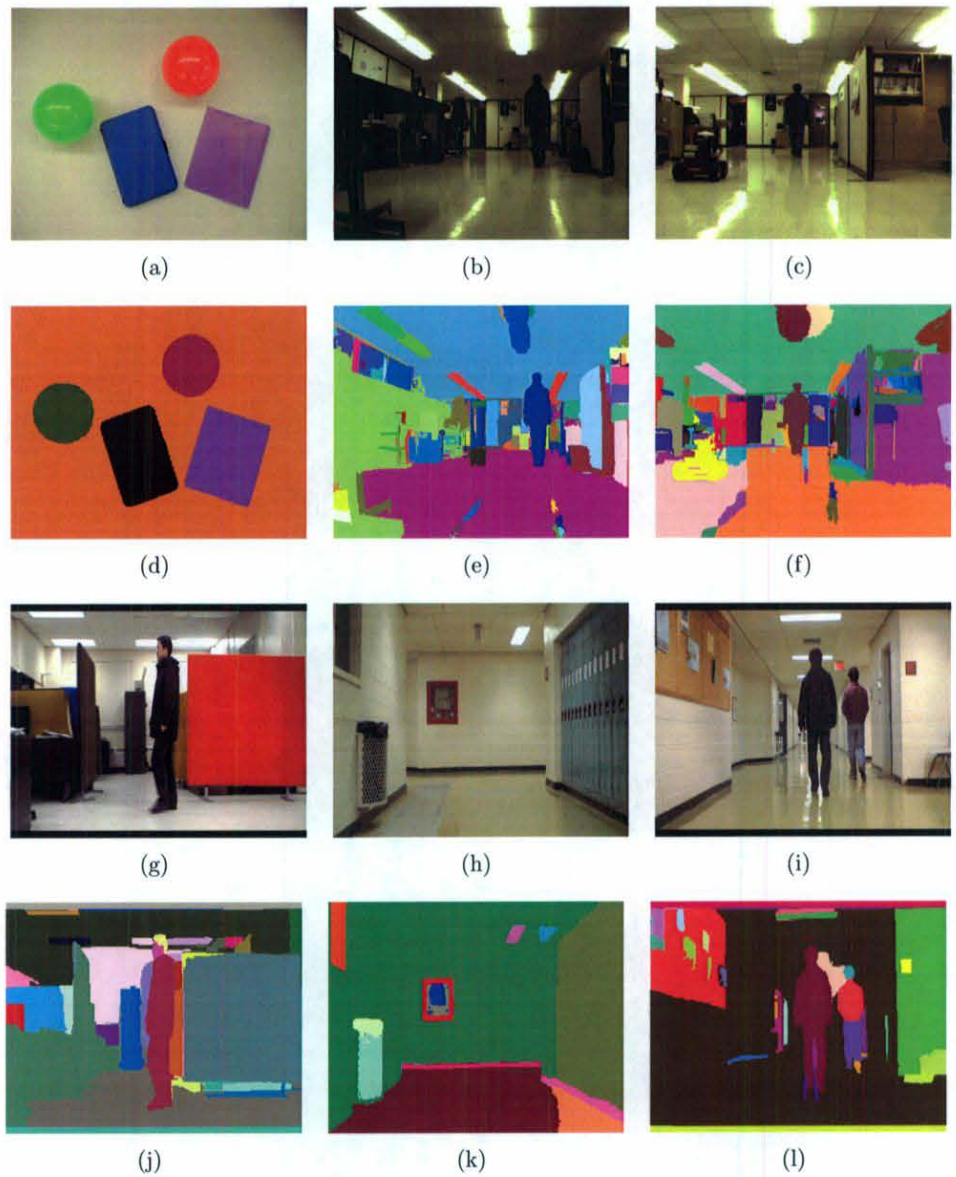


Figure 4.9: Results of pre-attentive segmentation in natural scenes. (a)-(c) and (g)-(i) Original images. (d)-(f) and (j)-(l) Pre-attentive segmentation results. Each color represents one proto-object.

where

$$M_{mn} = \sum_{\mathbf{r}_i} x^m y^n, \quad \forall \mathbf{r}_i \in \mathbf{R}_g, \tag{4.29}$$

where m and n denote the orders of moments with respect to X-axis and Y-axis, x and y are coordinates of the pixel $\mathbf{r}_i \in \mathbf{R}_g$.

4.3.7 Discussion

As shown in Figure 4.9, the proposed pre-attentive segmentation algorithm can successfully divide the complex scene into uniform proto-objects. For example, as shown in Figure 4.9(e), the chair (in the left side of the image) and the book (in the upper right side of the image) have been successfully segmented. A problem of this pre-attentive segmentation algorithm is that over-segmentation occurs in some cases, e.g., in the case that the lighting spreads unevenly on a single object. However, this is not a big problem since the perceptual completion of an object can be obtained through a top-down knowledge-based post-attentive perception procedure.

In fact, the Gestalt principle has illustrated that input stimuli and stored knowledge in LTM are two fundamental cues for perceptual grouping. It indicates that there are two ways for segmenting an image. One is unsupervised segmentation (i.e., bottom-up segmentation) based on input stimuli; the other is supervised segmentation (i.e., top-down segmentation) based on the stored knowledge.

Correspondingly, a few advanced algorithms that integrate bottom-up and top-down segmentation have been proposed. The first type of those algorithms, such as [147, 148], performs bottom-up segmentation and top-down detection simultaneously. The second type of those algorithms, such as [149], preforms bottom-up segmentation at first, followed by a serial top-down rectification on each segment. Compared with the first type, the advantage of the second type is that the expensive computation of top-down segmentation sequentially operates only on one region at a time, not over the whole image.

The proposed cognitive visual perception mechanism can also be used for image seg-

mentation by integrating pre-attentive segmentation, attentional selection and perceptual completion processing. Pre-attentive segmentation achieves proto-objects at first and then the perceptual completion procedure performs in cooperation with the serial attentional shift from one proto-object to others. This idea is close to the second type of algorithm presented above. The advantage of the proposed algorithm is that the attentional selection can give an optimal scan path (i.e., the sequence of proto-objects to be attended) for the serial top-down segmentation.

4.3.8 Computational Complexity

The computational complexity of this proposed pre-attentive segmentation algorithm can be approximately estimated as the sum of three parts across all pyramidal levels obtained during the accumulation process. Let n denote the number of vertices at a level l. The first part is the computational complexity of the decimation procedure. It can be denoted as $\mathcal{O}(N_d \times n)$, where N_d denotes the number of iterations of the decimation and $N_d \ll n$. The second part is the computational complexity of estimating the strength of inter-level edges and estimating of the aggregated features. It can be denoted as $\mathcal{O}(N_f \times n)$, where N_f denotes the number of features and $N_f = 3$. The third part is the computational complexity of the neighbor search procedure. It can be denoted as $\mathcal{O}((N_{nb} + N_{nb} \times N_{nb}) \times n)$, where N_{nb} denotes the number of neighbors of a vertex and $N_{nb} = 8$. The number n decreases approximately with the decimation ratio $1/4$ during the multi-scale accumulation. It can be seen that the computation of this algorithm is linear to the number n. The experiments have shown that the time cost of this algorithm for a 640×480 image is less than one second.

4.4 Conclusion

This chapter has presented the pre-attentive processing stage in the proposed cognitive visual perception paradigm. Two parts have been discussed in this chapter. The first

part is extraction of multi-scale pre-attentive features, including intensity, red-green pair, blue-yellow pair, orientation energy, contour and motion energy.

The second part of this chapter presents an algorithm for pre-attentive segmentation by extending the irregular pyramid technique. This algorithm is one of the main contributions of the proposed cognitive visual perception paradigm. There are several advantages of this proposed pre-attentive segmentation algorithm.

1. A probabilistic similarity measure (i.e., Bhattacharyya distance) is proposed for estimating the strength of intra-level edges in the irregular pyramid. Since each vertex at upper pyramidal levels is an accumulation of vertices at the levels below, this probabilistic similarity measure can precisely estimate the similarity between these accumulated vertices, compared with the deterministic similarity measures (e.g., Euclidean distance estimated by mean values) used in most of other pyramid-based segmentation methods. Furthermore, since this similarity measure is approximately scale-invariant during the pyramidal aggregation procedure, a constant similarity threshold can be used at multiple scales in the procedure of similarity-driven neighbor search.

2. A data-driven decimation routine is proposed. Compared with the SPD algorithm [140,142] used in other pyramid-based segmentation methods, this routine improves the segmentation performance in the sense that the vertices that can represent the neighbors as enough as possible deterministically survive during the decimation procedure, such that some of them can successfully emerge as final segments.

3. A new similarity-driven neighbor search method is proposed. In most of pyramid-based segmentation methods, the neighbor search procedure for the upper pyramidal levels is not presented clearly and its influence on the segmentation performance is extremely ignored. This proposed neighbor search procedure can improve segmentation precision by deterministically cutting connections between vertices that are located at a place with emergence of great transition. In other words, it pro-

vides a multi-scale way to accumulate the evidence of boundaries. As a result, the neighbors obtained at each pyramidal level are similar enough to the center vertex, such that the precision of final segmentation is improved.

Chapter 5

Attentional Selection

5.1 Introduction

The objective of attentional selection is to decide which proto-object should be attended by a combination of both bottom-up and top-down attention mechanisms. In order to realize this objective, four modules are required: bottom-up competition, top-down biasing, a combination of bottom-up competition and top-down biasing, as well as estimation of the proto-object based attentional activation. The bottom-up competition module yields a probabilistic location-based bottom-up saliency map and the top-down biasing module yields a probabilistic location-based top-down bias map. Once these two maps are probabilistically combined, a location-based attentional activation map is achieved. Based on the results of pre-attentive segmentation, a proto-object based attentional activation map is finally obtained. This chapter presents the detailed computational methods for realizing these four modules.

The first challenging issue in the attentional selection stage is the computational modeling of top-down biasing. According to the IC hypothesis [49], this issue includes several sub-problems: deduction of the task-relevant object given the task, deduction of the task-relevant feature(s) given the task-relevant object and estimation of the top-down biases given the task-relevant feature(s). The second challenging issue is the computational

modeling of the combination of bottom-up saliency and top-down biases at a uniform scale. This chapter presents approaches to these issues.

It is important to note that the saliency map and bias map are both estimated in a location-based manner. This is due to the fact that the extracted pre-attentive features are location-based, on which bottom-up competition and top-down biasing take place.

This chapter is organized as follows: Section 5.2 presents a bottom-up competition method by extending Itti's attention model [38]. Section 5.3 proposes a novel top-down biasing method based on the IC hypothesis. Section 5.4 proposes a method for combining bottom-up saliency and top-down biases in a probabilistic manner. Section 5.5 presents a method for estimating the proto-object based attentional activation map.

5.2 Bottom-up Competition

5.2.1 Background

The bottom-up competition module aims to produce the unconscious aspect of the proposed cognitive visual perception paradigm by modeling the bottom-up attention mechanism. The BC hypothesis [46] about the bottom-up attention mechanism posits that items in the scene compete for attention in terms of their conspicuousness in the spatial context, with a result that a salient item can be selected for attention. Thus how to estimate the conspicuousness of each item is the key point for modeling bottom-up attention. Conspicuousness represents the difference of an item from its spatial neighbors in terms of pre-attentive features. Accordingly, the estimation of conspicuousness consists of two components. The first component is *contrast* in terms of pre-attentive features in the spatial context, i.e., across multiple scales. It can be seen that the first component is consistent with the BC hypothesis. The second component is *integration* of these contrasts in terms of all pre-attentive features and across all spatial scales. It is obvious that the second component is consistent with the FIT [24].

Itti's model [38] has given an approach to both contrast and integration components,

finally yielding a location-based bottom-up saliency map. Thus, the bottom-up competition module proposed in this thesis is developed by extending Itti's model. Compared with Itti's model, two types of extensions are presented by the proposed bottom-up competition module. The first extension is that contour and motion features are included in the proposed bottom-up competition module. The second extension is that a new probabilistic representation of the location-based bottom-up saliency is proposed for further combination with the top-down biases.

The following three subsections respectively present the contrast algorithm, the integration algorithm and the estimation algorithm for the probabilistic location-based bottom-up saliency map.

5.2.2 Contrast

The contrast component can be modeled as calculating the difference between a pixel and its spatial neighbors in terms of pre-attentive features. Thus it can also be called *center-surround contrast*. The center is represented by the pixels at the fine scales (i.e., center scales) and its spatial neighbors are represented by the pixels at the coarse scales (i.e., surround scales). The center-surround contrast produces *center-surround difference maps* for each pre-attentive feature dimension. The implementation of center-surround contrast can be expressed as:

$$\mathbf{F}'_{int}(l_c, l_s) = |\mathbf{F}_{int}(l_c) \ominus \mathbf{F}_{int}(l_s)|, \tag{5.1}$$

$$\mathbf{F}'_{rg}(l_c, l_s) = |\mathbf{F}_{rg}(l_c) \ominus \mathbf{F}_{rg}(l_s)|, \tag{5.2}$$

$$\mathbf{F}'_{by}(l_c, l_s) = |\mathbf{F}_{by}(l_c) \ominus \mathbf{F}_{by}(l_s)|, \tag{5.3}$$

$$\mathbf{F}'_{o_\theta}(l_c, l_s) = |\mathbf{F}_{o_\theta}(l_c) \ominus \mathbf{F}_{o_\theta}(l_s)|, \tag{5.4}$$

$$\mathbf{F}'_{mv}(l_c, l_s) = |\mathbf{F}_{mv}(l_c) \ominus \mathbf{F}_{mv}(l_s)|, \tag{5.5}$$

where $l_c \in \{2, 3, 4\}$ and $l_s = l_c + \delta$ with $\delta \in \{3, 4\}$ respectively represent the center scales and surround scales, $\ominus$ denotes across-scale subtraction, $\theta \in \{0°, 45°, 90°, 135°\}$, and $\mathbf{F}'(l_c, l_s)$ denotes a center-surround difference map.

The across-scale subtraction operator $\ominus$ consists of two successive operations. The first one is interpolation, which interpolates the features at the surround scale to the center scale using the interpolation technique of the Gaussian pyramid as shown in (A.4) in Appendix A. The second operation is point-by-point subtraction between the interpolated features and the corresponding features at the center scale.

Examples of the center-surround contrast are shown in Figure 5.1.

5.2.3 Integration

It can be seen that six center-surround difference maps for each feature dimension are produced by the contrast algorithm and eight pre-attentive feature dimensions, i.e., intensity, red-green pair, blue-yellow pair, orientation energy in four preferred directions and motion energy are used. Thus totally $6 \times 8 = 48$ center-surround difference maps are obtained. The integration component combines all these center-surround difference maps to achieve conspicuity maps for each feature dimension. Finally, the integration component combines all conspicuity maps to achieve a location-based bottom-up saliency map. The integration component mainly consists of two operators, including normalization and across-scale addition, both of which are presented in the following subsections.

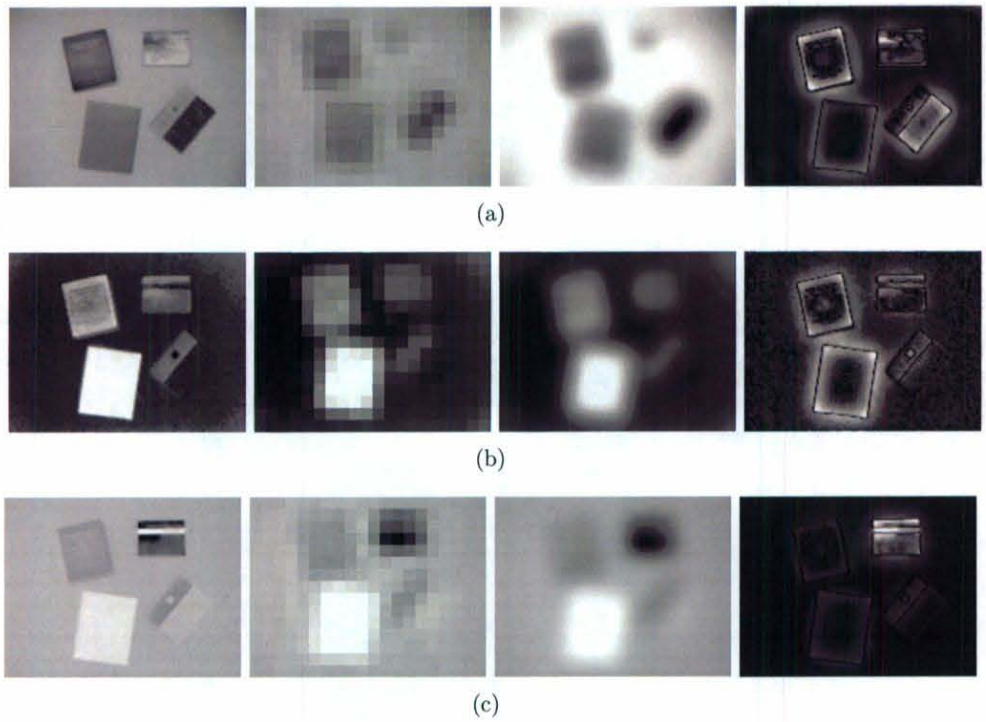


Figure 5.1: The center-surround contrast in terms of intensity, red-green pair and blue-yellow pair. Column 1: Pre-attentive feature at scale 2. Column 2: Pre-attentive feature at scale 5; In order to illustrate clearly, the images shown in column 2 are linearly scaled to the same size with the corresponding images in column 1. Column 3: Interpolation of the pre-attentive feature at scale 5 to scale 2. Column 4: Results of center-surround contrast, i.e., the center-surround difference maps $F'_f(2,5)$. Row 1: Center-surround contrast in terms of intensity. Row 2: Center-surround contrast in terms of red-green pair. Row 3: Center-surround contrast in terms of blue-yellow pair.

Normalization

Since the center-surround difference maps represent multiple modalities, the normalization of these difference maps is required before further integration. There are two objectives of the normalization operator. The first objective is to provide an approximately uniform scale to combine all difference maps. The second objective is to globally promote the salient items, so that the salient items that appear strongly in only a few difference maps cannot be masked by noise or by less salient items that present in a larger number of difference maps [38]. Thus, this normalization operator consists of two

successive steps:

1. Normalizing the values of each center-surround difference map into a fixed range $[0, 255]$. This is to satisfy the first objective by eliminating the influences of the modality-dependent amplitude.

2. Computing the average $\overline{m}$ of all local maxima in each difference map and globally multiplying the difference map by $(255-\overline{m})^2$. This is to satisfy the second objective by globally promoting the difference maps that contain a small number of strong peaks of activity (i.e., in the case that $(255-\overline{m})$ is large), while globally suppressing maps which contain numerous comparable peaks of activity (i.e., in the case that $(255-\overline{m})$ is small).

Across-scale Addition and Conspicuity Maps

These normalized center-surround difference maps of each pre-attentive feature dimension are combined to yield a *conspicuity map* of that feature dimension, which can be expressed as:

$$\mathbf{F}^s_{int} = \frac{1}{6} \bigoplus_{l_c=2}^{4} \bigoplus_{l_s=l_c+3}^{l_c+4} \mathcal{N}(\mathbf{F}'_{int}(l_c, l_s)), \tag{5.6}$$

$$\mathbf{F}^s_{rg} = \frac{1}{6} \bigoplus_{l_c=2}^{4} \bigoplus_{l_s=l_c+3}^{l_c+4} \mathcal{N}(\mathbf{F}'_{rg}(l_c, l_s)), \tag{5.7}$$

$$\mathbf{F}^s_{by} = \frac{1}{6} \bigoplus_{l_c=2}^{4} \bigoplus_{l_s=l_c+3}^{l_c+4} \mathcal{N}(\mathbf{F}'_{by}(l_c, l_s)), \tag{5.8}$$

$$\mathbf{F}^s_{o_\theta} = \frac{1}{6} \bigoplus_{l_c=2}^{4} \bigoplus_{l_s=l_c+3}^{l_c+4} \mathcal{N}(\mathbf{F}'_{o_\theta}(l_c, l_s)), \tag{5.9}$$

$$\mathbf{F}^s_{mv} = \frac{1}{6} \bigoplus_{l_c=2}^{4} \bigoplus_{l_s=l_c+3}^{l_c+4} \mathcal{N}(\mathbf{F}'_{mv}(l_c, l_s)), \tag{5.10}$$

where $\mathbf{F}^s$ denotes a conspicuity map, $\theta \in \{0°, 45°, 90°, 135°\}$, $\mathcal{N}(\cdot)$ is the normalization operator described above, $\bigoplus$ denotes across-scale addition.

The across-scale addition operator $\bigoplus$ consists of two operations: 1) interpolation of each normalized center-surround difference to scale 2 by using the interpolation technique of the Gaussian pyramid as shown in (A.4) in Appendix A, and 2) point-by-point addition.

***Conspicuity Map in terms of Contour*:** It can be seen that the contour feature is not shown in both contrast and integration components. In fact, the conspicuity map in terms of the contour feature can be estimated by a combination of the conspicuity maps in terms of the orientation energy in four preferred directions, based on the fact that the center-surround differences in terms of the contour feature can be approximately estimated by combining the center-surround differences in terms of orientation energy in four preferred directions. Therefore, estimation of the conspicuity map in terms of the contour feature can be expressed as:

$$\mathbf{F}^s_{ct} = \frac{1}{4} \sum_{\theta \in \{0°, 45°, 90°, 135°\}} \mathcal{N}(\mathbf{F}^s_{o_\theta}). \tag{5.11}$$

***Working Scale*:** It can be seen that all conspicuity maps are obtained at scale 2 and the following computation of the bottom-up saliency map also performs at scale 2. Thus, scale 2 is called *working scale* in this thesis.

Examples of conspicuity maps are shown in Figure 5.2.

Location-based Bottom-up Saliency Map

The obtained conspicuity maps in terms of all pre-attentive feature dimensions are finally combined to yield a location-based bottom-up saliency map. This combination can be expressed as:

$$\mathbf{S}_{bu} = \mathcal{N}(\mathbf{F}^s_{int}) + \frac{1}{2}\left[\mathcal{N}(\mathbf{F}^s_{rg}) + \mathcal{N}(\mathbf{F}^s_{by})\right] + \frac{1}{4}\left[\sum_{\theta} \mathcal{N}(\mathbf{F}^s_{o_\theta})\right] + \mathcal{N}(\mathbf{F}^s_{ct}) + \mathcal{N}(\mathbf{F}^s_{mv}), \tag{5.12}$$

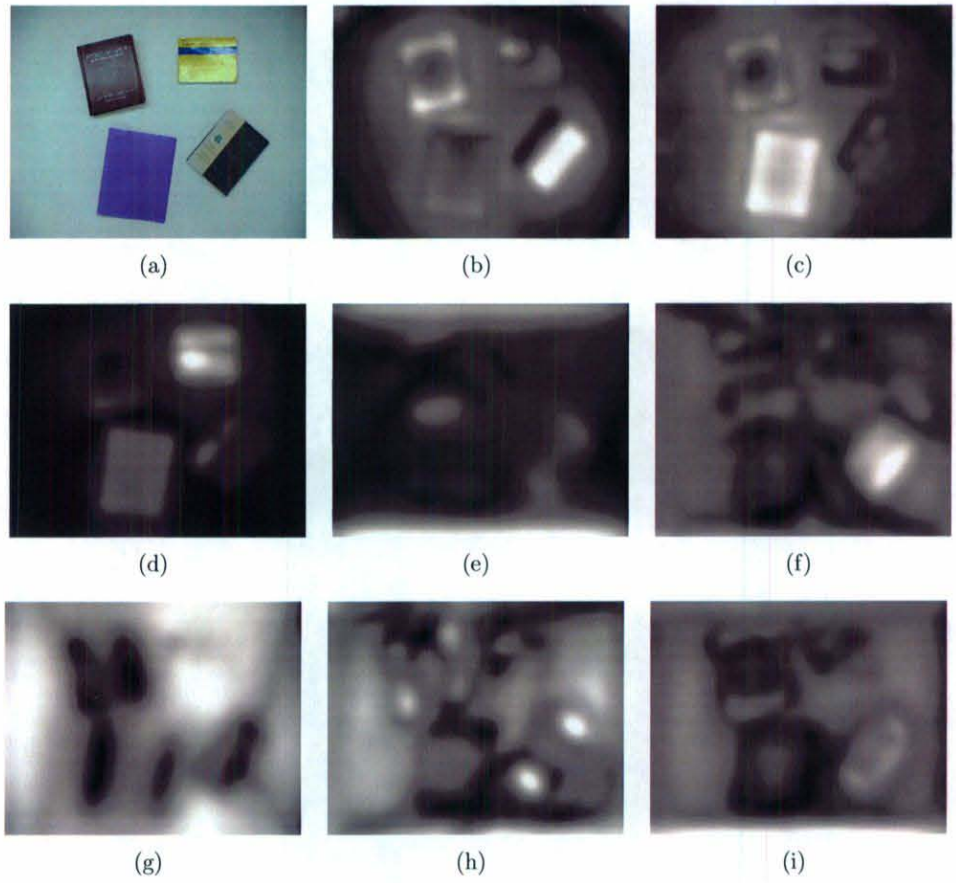


Figure 5.2: Conspicuity maps. (a) Original image. (b) Conspicuity map in terms of intensity. (c) Conspicuity map in terms of red-green pair. (d) Conspicuity map in terms of blue-yellow pair. (e)-(h) Conspicuity map in terms of local orientation energy in four preferred directions. (i) Conspicuity map in terms of contour.

where $\mathbf{S}_{bu}$ denotes the location-based bottom-up saliency map.

5.2.4 Probabilistic Representation of Bottom-up Saliency

If only bottom-up attention is used to guide the focus of attention, the bottom-up saliency map obtained in (5.12) would be sufficient to guide attentional selection. However, the integration of top-down attention will lead to a challenging problem, which is how the bottom-up saliency and top-down biases are combined at a uniform scale. This thesis

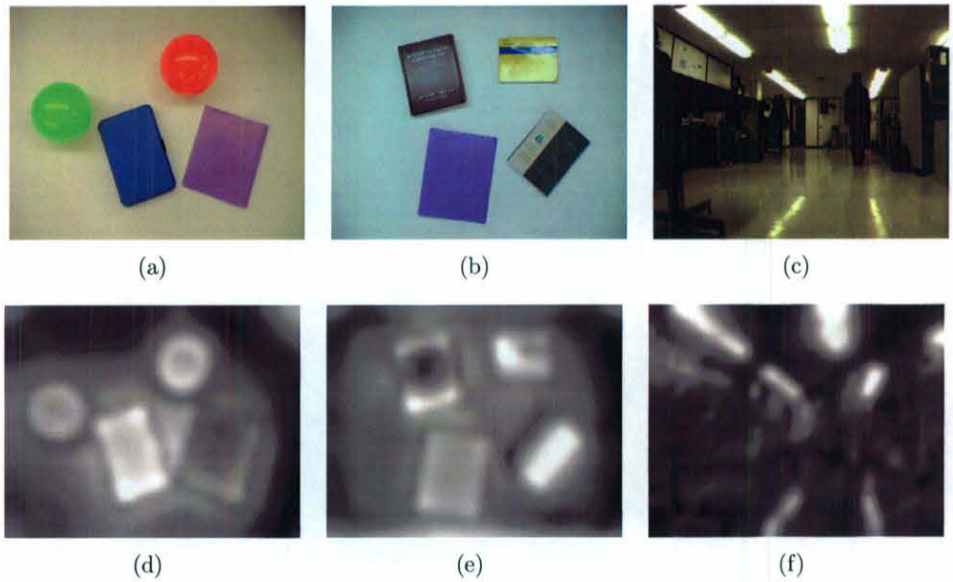


Figure 5.3: Probabilistic location-based bottom-up saliency maps. (a)-(c) Original images. (d)-(f) Probabilistic location-based bottom-up saliency maps. The maps are normalized to [0,255] for display.

models top-down attention as a Bayesian probabilistic procedure, which will be presented in section 5.3. Thus, a probabilistic representation of the location-based bottom-up saliency is required, so that a probabilistic method, as shown in section 5.4, can be proposed to combine the bottom-up saliency and top-down biases.

Given the following two assumptions: 1) the selection process guided by the space-based bottom-up attention is a random event, and 2) the sample space of this random event is composed of all spatial locations in the image, the salience of a spatial location can be used to represent the degree of belief that bottom-up attention selects that location. Therefore, the probability of a spatial location $\mathbf{r}_i$ being attended by the bottom-up attention mechanism can be estimated using probability theory:

$$p_{bu}(\mathbf{r}_i) = \frac{S_{bu}(\mathbf{r}_i)}{\sum_{\mathbf{r}_{i'} \in \mathbf{I}} S_{bu}(\mathbf{r}_{i'})}, \tag{5.13}$$

where $p_{bu}(\mathbf{r}_i)$ denotes the probability of a spatial location $\mathbf{r}_i$ being attended by the

bottom-up attention mechanism, the denominator $\sum_{\mathbf{r}_{i'} \in \mathbf{I}} S_{bu}(\mathbf{r}_{i'})$ is the normalizing constant and $\mathbf{I}$ denotes the input image.

Some examples of probabilistic location-based bottom-up saliency maps are shown in Figure 5.3.

5.3 Top-down Biasing

5.3.1 Background

The top-down biasing module aims to produce the conscious aspect of the proposed cognitive visual perception paradigm by modeling the top-down attention mechanism. The top-down attention mechanism is a conscious, task-driven way to guide the focus of attention to a task-relevant object. Although modeling the top-down attention mechanism is still in development, this thesis posits that object-based top-down attention is mainly influenced by two factors: the current task and LTM object representations.

The robotic task addressed by this thesis refers to a specification of the object, on which an action executes. The object specified by the task is termed as a *task-relevant object* in this thesis. In other words, the task-relevant object refers to an object whose occurrence is expected by the current task. Based on this definition, a task-relevant object can be directly or indirectly obtained from the current task. Therefore the factor of the current task leads to an issue of deducing the task-relevant object from the current task.

Given the task-relevant object, its LTM representation can be recalled from LTM. Then this recalled LTM object representation can be used to estimate the possibility of each item in the current scene belonging to an instance of the task-relevant object. Therefore the factor of LTM object representations leads to the issue of estimating the top-down bias given a task-relevant object. This thesis proposes that Duncan's IC hypothesis [49] can be used for solving this issue. The related aspect of the IC hypothesis for guiding the estimation of top-down biases can be summarized as follows: by direct-

ing attention to a conspicuous cue of an object, a competitive advantage over the whole object is produced. This indicates that the top-down biases can be estimated only using the conspicuous cue of the task-relevant object. This thesis terms the conspicuous cue of the task-relevant object as a *task-relevant feature*. This thesis further presents that the task-relevant feature can be deduced from the learned LTM representation of the task-relevant object. Since task-relevant features are conspicuous and low-level, this proposed top-down biasing method is effective and efficient.

Based on the above discussion, the proposed top-down biasing module consists of four steps:

1. Deduction of a task-relevant object from the task.
2. Deduction of task-relevant feature(s) from the task-relevant object: The learned LTM representation of the task-relevant object is recalled from LTM to deduce one or a few task-relevant feature(s).
3. Construction of the attentional template(s) in WM using the task-relevant feature(s).
4. Estimation of location-based top-down biases: A location-based top-down bias map is estimated by comparing attentional template(s) with corresponding pre-attentive feature(s).

The following subsections respectively present the deduction of a task-relevant object, the structure of LTM object representations, deduction of task-relevant feature(s), construction of attentional template(s) and estimation of top-down biases.

5.3.2 Robotic Tasks and Task-relevant Objects

This thesis proposes that robotic tasks can be grouped into two categories.

Type I Tasks

In the first category, the task directly specifies the task-relevant object at a moment, e.g., searching for an apple. This category of tasks is called *Type I* in this thesis.

Type II Tasks

In the second category, the task does not directly specify the task-relevant object, e.g., navigation. However, this thesis proposes that the task-relevant object (e.g., a landmark) can also be deduced based on the learned cognitive perception-action mapping. This category of tasks is called *Type II* in this thesis.

Cognitive perception-action mapping related to Type II tasks can be briefly discussed as follows. Cognitive perception-action mapping can be generally defined as an association between perception, context and actions. According to the proposed cognitive visual perception paradigm, the cognitive perception-action mapping can be modeled as an association between attentional states, context and actions. Consistent with recent research in the area of cognitive robots (e.g., AMD [113, 114]), this thesis proposes that actions of a cognitive robot can be categorized into two types. The first type is external actions, which guide the operation of effectors. The second type is internal actions, which mainly includes guidance for attentional selection at the next moment. Thus, the cognitive mapping related to Type II tasks is the association between the current attentional state and the next possible attentional state (i.e., attentional prediction). Since the proposed cognitive visual perception paradigm is object-based, the attentional state is an instance of the object that is attended at the current moment and the attentional prediction is an instance of the task-relevant object at the next moment. It can be seen that a Type II task corresponds to a learned cognitive perception-action mapping.

It can be further proposed that the cognitive perception-action mapping used to represent a Type II task can be modeled by using a first-order discrete Markov process (FDMP). The FDMP can be expressed as $p(a_{t+1}|a_t)$, where a_t denotes the attentional state at moment t and a_{t+1} denotes the attentional state at moment $t+1$ (i.e. attentional

prediction). This definition means that the probability of each attentional state prediction for the next moment can be estimated given the attentional state at the current moment. In the proposed object-based cognitive perception paradigm, a set of discrete attentional states is composed of the LTM object representations based on the fact that an LTM object representation encodes a couple of instances of that object. Constructing the cognitive perception-action mapping related to Type II tasks includes two factors: 1) learning of LTM object representations, and 2) learning of the association modeled by FDMP. Since this thesis only considers the perception mechanism, the learning of LTM object representations is presented in section 6.4 in Chapter 6, whereas the learning of association is considered as outside the scope of this thesis.

The conclusion of this subsection is that the task-relevant object can also be deduced from Type II tasks.

5.3.3 Structure of Object Representations Related to Top-down Biasing

Once the task-relevant object is obtained from the current task, its LTM object representation is then recalled from LTM to deduce the task-relevant feature(s). Although the construction and learning of LTM object representations are carried out in the post-attentive perception stage, a brief description of the structure of LTM object representations is given in this subsection in order to clearly present the deduction of the task-relevant feature(s). It is important to note that this chapter only presents the structure of the LTM object representations related to the top-down biasing module. The complete structure and learning algorithms of the LTM object representations will be presented in section 6.4 in Chapter 6.

Dual-coding Structure

The proposed LTM object representation $\mathbf{O}$ includes two codings: the global coding $\mathbf{O}_{gb}$ and the local coding $\mathbf{O}_{lc}$. The global coding is built using the contour feature, whereas

the local coding is built using intensity, color and local orientations.

Each coding includes two descriptors: appearance $\mathbf{O}^a$ and salience $\mathbf{O}^s$. The appearance descriptor represents the appearance value of each feature dimension. The salience descriptor represents conspicuousness of each feature dimension and it is used to deduce the task-relevant feature(s) at the beginning of top-down biasing.

The proposed dual-coding LTM object representation can be expressed as:

$$\mathbf{O} = \begin{pmatrix} \mathbf{O}^a_{ct} & \mathbf{O}^a_{int} & \mathbf{O}^a_{rg} & \mathbf{O}^a_{by} & \mathbf{O}^a_{o_{0^\circ}} & \mathbf{O}^a_{o_{45^\circ}} & \mathbf{O}^a_{o_{90^\circ}} & \mathbf{O}^a_{o_{135^\circ}} \\ \mathbf{O}^s_{ct} & \mathbf{O}^s_{int} & \mathbf{O}^s_{rg} & \mathbf{O}^s_{by} & \mathbf{O}^s_{o_{0^\circ}} & \mathbf{O}^s_{o_{45^\circ}} & \mathbf{O}^s_{o_{90^\circ}} & \mathbf{O}^s_{o_{135^\circ}} \end{pmatrix}, \tag{5.14}$$

where $\mathbf{O}_{ct}$ is the global coding in terms of contour, $\mathbf{O}_{int}$, $\mathbf{O}_{rg}$, $\mathbf{O}_{by}$, $\mathbf{O}_{o_{0^\circ}}$, $\mathbf{O}_{o_{45^\circ}}$, $\mathbf{O}_{o_{90^\circ}}$ and $\mathbf{O}_{o_{135^\circ}}$ respectively represent local codings in terms of intensity, red-green pair, blue-yellow pair and local orientations in four directions.

Statistical Structure

The proposed LTM object representation is built in a statistical form in the sense that it can statistically encode a couple of instances of that object obtained from different views and under different conditions.

Statistical Global Appearance Descriptor: In this thesis, the B-Spline technique [150,151] is employed to represent a contour. In other words, a contour is represented by using a set of control points along the contour curve. Since a variety of contours could be present from different views for an object, especially for a 3-dimensional object, the global coding of an object is composed of a set of contours observed from multiple views. Each entry of this set of contours is termed as a *contour instance* in this thesis. Therefore the global appearance descriptor can be expressed as:

$$\mathbf{O}^a_{ct} = \begin{pmatrix} \mathbf{O}^{a,1}_{ct} & \mathbf{O}^{a,2}_{ct} & \cdots & \mathbf{O}^{a,N_{ct}}_{ct} \end{pmatrix}, \tag{5.15}$$

where N_{ct} denotes the number of contour instances in the global coding of an object.

The statistical structure of a contour instance in the global appearance descriptor can be expressed as,

$$\mathbf{O}_{ct}^{a,n} = \begin{pmatrix} \mu_{x,n}^{a,1} & \cdots & \mu_{x,n}^{a,N_{cp}^n} & \mu_{y,n}^{a,1} & \cdots & \mu_{y,n}^{a,N_{cp}^n} \\ \sigma_{x,n}^{a,1} & \cdots & \sigma_{x,n}^{a,N_{cp}^n} & \sigma_{y,n}^{a,1} & \cdots & \sigma_{y,n}^{a,N_{cp}^n} \end{pmatrix}, \tag{5.16}$$

where μ_x^a and μ_y^a denote the mean in terms of the spatial position of a control point along the contour instance, σ_x^a and σ_y^a denote the standard deviation (STD) in terms of the spatial position of a control point, $n \in \{1, 2, ..., N_{ct}\}$ is the index of the contour instance of that object, and N_{cp}^n denotes the number of control points along the contour instance indexed by n.

Statistical Local Appearance Descriptors: In the proposed LTM object representation, the local coding of an object is composed of local parts of that object. Low-order statistics, including the means and STDs in terms of intensity, red-green pair, blue-yellow pair and local orientations are used to represent the appearance of each part belonging to the object. These low-order statistics constitute the statistical local appearance descriptors, which can be expressed as:

$$\mathbf{O}_{int}^{a} = \begin{pmatrix} \mu_{int}^{a,1} & \mu_{int}^{a,2} & \cdots & \mu_{int}^{a,N_p} \\ \sigma_{int}^{a,1} & \sigma_{int}^{a,2} & \cdots & \sigma_{int}^{a,N_p} \end{pmatrix}, \tag{5.17}$$

$$\mathbf{O}_{rg}^{a} = \begin{pmatrix} \mu_{rg}^{a,1} & \mu_{rg}^{a,2} & \cdots & \mu_{rg}^{a,N_p} \\ \sigma_{rg}^{a,1} & \sigma_{rg}^{a,2} & \cdots & \sigma_{rg}^{a,N_p} \end{pmatrix}, \tag{5.18}$$

$$\mathbf{O}_{by}^{a} = \begin{pmatrix} \mu_{by}^{a,1} & \mu_{by}^{a,2} & \cdots & \mu_{by}^{a,N_p} \\ \sigma_{by}^{a,1} & \sigma_{by}^{a,2} & \cdots & \sigma_{by}^{a,N_p} \end{pmatrix}, \tag{5.19}$$

$$\mathbf{O}^{a}_{o_\theta} = \begin{pmatrix} \mu^{a,1}_{o_\theta} & \mu^{a,2}_{o_\theta} & \cdots & \mu^{a,N_p}_{o_\theta} \\ \sigma^{a,1}_{o_\theta} & \sigma^{a,2}_{o_\theta} & \cdots & \sigma^{a,N_p}_{o_\theta} \end{pmatrix}, \tag{5.20}$$

where μ^a and σ^a respectively denote the mean and STD of appearance values in terms of the corresponding feature dimension of a part, N_p is the number of parts of that object, and $\theta \in \{0°, 45°, 90°, 135°\}$.

***Statistical Salience Descriptors*:** The statistics, including the mean and STD, of the conspicuity (calculated in (5.6), (5.7), (5.8), (5.9) and (5.11) respectively) in terms of each feature dimension are used to build the salience descriptors in the proposed LTM object representation.

The statistical global salience descriptor can be expressed as:

$$\mathbf{O}^{s}_{ct} = \begin{pmatrix} \mu^{s,1}_{ct} & \mu^{s,2}_{ct} & \cdots & \mu^{s,N_{ct}}_{ct} \\ \sigma^{s,1}_{ct} & \sigma^{s,2}_{ct} & \cdots & \sigma^{s,N_{ct}}_{ct} \end{pmatrix} \tag{5.21}$$

where μ^s_{ct} and σ^s_{ct} respectively denote the mean and STD of the conspicuity values in terms of the contour feature over all control points along a contour instance.

The statistical local salience descriptors can be expressed as:

$$\mathbf{O}^{s}_{int} = \begin{pmatrix} \mu^{s,1}_{int} & \mu^{s,2}_{int} & \cdots & \mu^{s,N_p}_{int} \\ \sigma^{s,1}_{int} & \sigma^{s,2}_{int} & \cdots & \sigma^{s,N_p}_{int} \end{pmatrix} \tag{5.22}$$

$$\mathbf{O}^{s}_{rg} = \begin{pmatrix} \mu^{s,1}_{rg} & \mu^{s,2}_{rg} & \cdots & \mu^{s,N_p}_{rg} \\ \sigma^{s,1}_{rg} & \sigma^{s,2}_{rg} & \cdots & \sigma^{s,N_p}_{rg} \end{pmatrix} \tag{5.23}$$

$$\mathbf{O}^{s}_{by} = \begin{pmatrix} \mu^{s,1}_{by} & \mu^{s,2}_{by} & \cdots & \mu^{s,N_p}_{by} \\ \sigma^{s,1}_{by} & \sigma^{s,2}_{by} & \cdots & \sigma^{s,N_p}_{by} \end{pmatrix} \tag{5.24}$$

$$\mathbf{O}_{o_\theta}^{s} = \begin{pmatrix} \mu_{o_\theta}^{s,1} & \mu_{o_\theta}^{s,2} & \cdots & \mu_{o_\theta}^{s,N_p} \\ \sigma_{o_\theta}^{s,1} & \sigma_{o_\theta}^{s,2} & \cdots & \sigma_{o_\theta}^{s,N_p} \end{pmatrix} \tag{5.25}$$

where μ^s and σ^s respectively denote the mean and STD of conspicuity values in terms of the corresponding local features of a part, and $\theta \in \{0°, 45°, 90°, 135°\}$.

5.3.4 Task-relevant Feature(s)

Once the task-relevant object is deduced, the salience descriptors in the LTM representation of that object are firstly recalled from LTM and used to deduce the task-relevant feature dimension(s). This deduction is implemented by finding the feature dimension(s) that have greater conspicuity. Therefore this deduction process can be expressed as:

$$(f_{rel}, n_{rel}) = \arg \max_{f \in \{ct, int, rg, by, o_\theta\}} \max_{n \in \{1,2,\ldots,N\}} \frac{\mu_f^{s,n}}{1 + \sigma_f^{s,n}}, \tag{5.26}$$

where N denotes the number of parts or contour instances in the LTM representation of the task-relevant object, i.e., $N = N_{ct}$ when $f = ct$, whereas $N = N_p$ when $f \in \{int, rg, by, o_\theta\}$; μ^s and σ^s respectively denote the mean and STD of salience descriptors in the LTM representation of the task-relevant object, f_{rel} denotes the *task-relevant feature dimension(s)*, and n_{rel} denotes the index of the *task-relevant part* or the index of the *task-relevant contour instance*.

The term $\frac{\mu_f^{s,n}}{1+\sigma_f^{s,n}}$ in (5.26) is called *task-relevance* in this thesis.

The proposed cognitive perception paradigm first selects only one task-relevant feature dimension (i.e., the most conspicuous dimension) given the task-relevant object. If post-attentive perception shows that the result of attentional selection is incorrect, the proposed paradigm selects more task-relevant feature dimensions (i.e., several top conspicuous dimensions) to guide attentional selection again.

The Type I task can also specify the task-relevant feature directly. For instance, if the task is to search for a vertically aligned object in the present scene, then orientation

in 90° will be the task-relevant feature. If the task is to search for a moving object, then motion will be the task-relevant feature.

It can be seen that nine feature dimensions can be used as candidates of task-relevant feature(s) in the proposed perception paradigm: they are contour, intensity, red-green pair, blue-yellow pair, orientation in 0°, orientation in 45°, orientation in 90°, orientation in 135° and motion.

There are two advantages of the proposed method for deducing the task-relevant feature(s). The first advantage is effectiveness due to the conspicuousness of the task-relevant feature(s). The second advantage is efficiency since the task-relevant feature(s) are low-level.

5.3.5 Attentional Template(s)

Once the task-relevant feature dimension(s) are deduced, the appearance descriptors in terms of the task-relevant feature dimension(s) in the LTM representation of the task-relevant object are recalled from LTM. The appearance descriptors are thereby called *task-relevant feature(s)*. The task-relevant feature in each dimension is used to form an attentional template [152] in WM in order to estimate the top-down bias in that dimension. Note that $\mathbf{F}^t$ is used to denote the attentional template in this thesis.

If f_{rel} is contour, the global appearance descriptor of the task-relevant contour instance, as shown in (5.16), is recalled from LTM to build an attentional template $\mathbf{F}^t_{ct}$ in WM:

$$\mathbf{F}^t_{ct} = \begin{pmatrix} \mathbf{F}^{t,\mu}_{ct} & \mathbf{F}^{t,\sigma}_{ct} \end{pmatrix} = (\mathbf{O}^{a,n_{rel}}_{ct})^T. \tag{5.27}$$

It can be further written as:

$$\begin{pmatrix} \mathbf{F}^{t,\mu}_{ct} & \mathbf{F}^{t,\sigma}_{ct} \end{pmatrix} = \begin{pmatrix} \mathbf{F}^{t,\mu,x}_{ct} & \mathbf{F}^{t,\sigma,x}_{ct} \\ \mathbf{F}^{t,\mu,y}_{ct} & \mathbf{F}^{t,\sigma,y}_{ct} \end{pmatrix} = \begin{pmatrix} \mu^{a,1}_{x,n_{rel}}, \ldots, \mu^{a,N^{n_{rel}}_{cp}}_{x,n_{rel}}, \mu^{a,1}_{y,n_{rel}}, \ldots, \mu^{a,N^{n_{rel}}_{cp}}_{y,n_{rel}} \\ \sigma^{a,1}_{x,n_{rel}}, \ldots, \sigma^{a,N^{n_{rel}}_{cp}}_{x,n_{rel}}, \sigma^{a,1}_{y,n_{rel}}, \ldots, \sigma^{a,N^{n_{rel}}_{cp}}_{y,n_{rel}} \end{pmatrix}^T. \tag{5.28}$$

If f_{rel} is intensity, red-green pair or blue-yellow pair, the local appearance descriptor

of the task-relevant part in terms of the corresponding feature dimension in the LTM object representation, as shown in (5.17), (5.18) and (5.19) respectively, is recalled from LTM to build an attentional template $\mathbf{F}_{int}^{t}$, $\mathbf{F}_{rg}^{t}$ or $\mathbf{F}_{by}^{t}$ in WM:

$$\mathbf{F}_{int}^{t} = \left(\begin{array}{cc} F_{int}^{t,\mu} & F_{int}^{t,\sigma} \end{array} \right) = \left(\begin{array}{cc} \mu_{int}^{a,n_{rel}} & \sigma_{int}^{a,n_{rel}} \end{array} \right), \tag{5.29}$$

$$\mathbf{F}_{rg}^{t} = \left(\begin{array}{cc} F_{rg}^{t,\mu} & F_{rg}^{t,\sigma} \end{array} \right) = \left(\begin{array}{cc} \mu_{rg}^{a,n_{rel}} & \sigma_{rg}^{a,n_{rel}} \end{array} \right), \tag{5.30}$$

$$\mathbf{F}_{by}^{t} = \left(\begin{array}{cc} F_{by}^{t,\mu} & F_{by}^{t,\sigma} \end{array} \right) = \left(\begin{array}{cc} \mu_{by}^{a,n_{rel}} & \sigma_{by}^{a,n_{rel}} \end{array} \right). \tag{5.31}$$

If f_{rel} is the local orientation in a direction θ, it indicates that the task-relevant part should be present in that direction in the current scene. Thus an attentional template F_{o}^{t} is built in WM by directly using θ. It can be expressed as:

$$F_{o}^{t} = \theta. \tag{5.32}$$

If the motion is specified as f_{rel}, an attentional template F_{mv}^{t} is built in WM and is set to 1:

$$F_{mv}^{t} = \begin{cases} 1 & \text{if motion is task-relevant} \\ 0 & \text{otherwise} \end{cases}. \tag{5.33}$$

It can be seen that the attentional templates in terms of most feature dimensions are low-order statistics (i.e., the mean and STD) obtained from the LTM object representation that is developed statistically by encoding a couple of training instances of that object at previous moments.

5.3.6 Estimation of Location-based Top-down Biases

Estimation of location-based top-down biases is a comparison procedure between the attentional template and corresponding pre-attentive feature extracted from the input at the working scale.

The Framework of Probabilistic Top-Down Biasing

Based on the fact that most of the attentional templates are in a statistical form, Bayesian inference is therefore the best way to estimate the location-based top-down bias, which represents the probability of a spatial location being an instance of the task-relevant object. The advantage of this probabilistic approach is the robustness to perceptual uncertainties.

The proposed probabilistic top-down biasing approach can be generally expressed by using Bayes' theorem:

$$p_{td}(\mathbf{r}_i|\mathbf{F}^t) = \frac{p_{td}(\mathbf{F}^t|\mathbf{r}_i) \times p_{td}(\mathbf{r}_i)}{\sum_{\mathbf{r}_{i'} \in \mathbf{I}} p_{td}(\mathbf{F}^t|\mathbf{r}_{i'}) \times p_{td}(\mathbf{r}_{i'})}, \tag{5.34}$$

where $p_{td}(\mathbf{r}_i)$ denotes the prior probability of a location $\mathbf{r}_i$ being attended by the top-down attention mechanism, $p_{td}(\mathbf{F}^t|\mathbf{r}_i)$ denotes the observation likelihood, $p_{td}(\mathbf{r}_i|\mathbf{F}^t)$ is the posterior probability of the location $\mathbf{r}_i$ being attended by the top-down attention mechanism, and the denominator $\sum_{\mathbf{r}_{i'} \in \mathbf{I}} p_{td}(\mathbf{F}^t|\mathbf{r}_{i'}) \times p_{td}(\mathbf{r}_{i'})$ is the normalizing factor.

Since the prior probability $p_{td}(\mathbf{r}_i)$ can be seen as a prediction before the attentional template is formed, $p_{td}(\mathbf{r}_i)$ is assumed to be a uniform distribution. Alternatively, the posterior distribution at the last moment can be regarded as the prior distribution at the current moment if the attentional deployment is modeled as a dynamical Markov process. However, this thesis only focuses on the research of attentional deployment without the influences caused in the temporal context. Based on the assumption that each location has the same probability to be attended by the top-down attention mechanism before observation, the prior $p_{td}(\mathbf{r}_i)$ is therefore modeled by using a uniform distribution in this

thesis.

Given the assumption about the prior distribution, top-down biasing can be simplified into the problem of estimating the observation likelihood $p_{td}(\mathbf{F}^t|\mathbf{r}_i)$. The following will present the detailed implementation of estimating $p_{td}(\mathbf{F}^t|\mathbf{r}_i)$ for each type of task-relevant feature.

Biasing in terms of Contour

The contour is a type of global feature. In other words, each proto-object $\mathbf{R}_g$ obtained from the pre-attentive segmentation is the basic unit of top-down biasing in terms of contour. Therefore, the observation likelihoods in terms of contour at locations within the same proto-object are identical, i.e., $p_{td}(\mathbf{F}^t_{ct}|\mathbf{r}_i) = p_{td}(\mathbf{F}^t_{ct}|\mathbf{R}_g)$, if $\mathbf{r}_i \in \mathbf{R}_g$.

If the task-relevant object includes multiple parts (the number of parts is N_p), N_p neighbor proto-objects of a working proto-object are combined as a unified temporary grouping to evaluate the bias of that working proto-object. In the following paragraphs, the term *proto-object* is still used to represent a single proto-object or the unified temporary grouping.

Due to the non-rigidity of contours, the accurate alignment of the contour specified by the attentional template to a proto-object is difficult to achieve. Thus, a set of predicted contours estimated from the attentional template is required first for each proto-object. The result is that the estimation of $p_{td}(\mathbf{F}^t_{ct}|\mathbf{R}_g)$ can be modeled as another set of Bayesian inference processes in the sense that the posterior probability obtained based on a predicted contour can be used as a candidate estimation of $p_{td}(\mathbf{F}^t_{ct}|\mathbf{R}_g)$. Finally, maximum a posteriori (MAP) estimation is used to select one candidate as the final estimation of $p_{td}(\mathbf{F}^t_{ct}|\mathbf{R}_g)$. That is, the maximal one among all candidates is selected as the observation likelihood of that proto-object.

Five steps are included in the estimation of $p_{td}(\mathbf{F}^t_{ct}|\mathbf{R}_g)$.

Step 1: In this thesis, a contour curve is represented by extending the active contour technique and B-Spline technique [150, 151]. Thus, a contour curve can be defined as:

$$\mathbf{C} = f_c(\mathbf{W}\mathbf{X} + \mathbf{Q}_0), \tag{5.35}$$

where $\mathbf{C}$ denotes a contour curve, f_c and $\mathbf{W}$ represent B-Spline basis functions and the shape factor respectively, both of which are fixed given $\mathbf{Q}_0$ and the definitions of which will be given in (5.40) and (5.41) respectively, $\mathbf{X}$ is the shape state vector, and $\mathbf{Q}_0$ is the control point vector that can be expressed as:

$$\mathbf{Q}_0 = \begin{pmatrix} \mathbf{Q}_0^x \\ \mathbf{Q}_0^y \end{pmatrix} = \begin{pmatrix} x_1, x_2, ..., x_P \\ y_1, y_2, ..., y_P \end{pmatrix}, \tag{5.36}$$

where (x_1, y_1), (x_2, y_2),...,(x_P, y_P) are coordinates of control points along the contour $\mathbf{C}$, and P is the number of control points.

Control point vector $\mathbf{Q}_0$ characterizes the object's basic shape and thereby it can be used for shape discrimination between the task-specific object and distractors. The shape state vector $\mathbf{X}$ represents the spatial transformation of a contour instance $\mathbf{C}$ with respect to $\mathbf{Q}_0$. Using both $\mathbf{Q}_0$ and $\mathbf{X}$, the shape of a rigid object or a simple non-rigid object can be described. In other words, a contour can be determined by the state vector $\mathbf{X}$ and the control point vector $\mathbf{Q}_0$.

It can be seen that the attentional template is used as the control point vector, i.e., $\mathbf{Q}_0 = \mathbf{F}_{ct}^{t,\mu}$, when estimating the top-down bias in terms of contour. The shape state vector $\mathbf{X}$ includes six elements as shown in (5.37). Translation is determined by the first two entries x_1 and x_2, scaling is mainly controlled by the middle two entries x_3 and x_4, and rotation is controlled by all the last four entries from x_3 to x_6.

$$\mathbf{X} = \begin{pmatrix} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 \end{pmatrix}^T. \tag{5.37}$$

Thus, the first step is to predict a set of prior shape states $\{\mathbf{X}^m_{\mathbf{R}_g}\}$ for each proto-object, where $m \in \{1, 2, \ldots, N_c\}$ and N_c is the number of prior shape states. According to our experiments, N_c is empirically set to 200 to achieve a good balance between satisfactory performance and computational cost.

At first, a deterministic prediction state $\mathbf{X}'_{\mathbf{R}_g}$ is calculated based on the attentional template and properties of the proto-object $\mathbf{R}_g$. The objective of calculating $\mathbf{X}'_{\mathbf{R}_g}$ is to approximately align the contour represented by the attentional template to the proto-object $\mathbf{R}_g$. Thus $\mathbf{X}'_{\mathbf{R}_g}$ can be obtained using the homogeneous transformation:

$$\begin{aligned}
\mathbf{X}'_{\mathbf{R}_g} &= \begin{pmatrix} x'_1 & x'_2 & x'_3 & x'_4 & x'_5 & x'_6 \end{pmatrix}^T \\
x'_1 &= c^x_{\mathbf{R}_g} - c^x_{ct} \\
x'_2 &= c^y_{\mathbf{R}_g} - c^y_{ct} \\
x'_3 &= x'_4 = \sqrt{A_{\mathbf{R}_g}/A_{ct}}\cos\theta_d - 1 \\
x'_5 &= -x'_6 = \sqrt{A_{\mathbf{R}_g}/A_{ct}}\sin\theta_d
\end{aligned} \tag{5.38}$$

where $(c^x_{\mathbf{R}_g}, c^y_{\mathbf{R}_g})$ are centroid coordinates of the g_{th} proto-object $\mathbf{R}_g$, (c^x_{ct}, c^y_{ct}) are centroid coordinates of the contour represented by the attentional template $\mathbf{F}^{t,\mu}_{ct}$, $A_{\mathbf{R}_g}$ is the area of the g_{th} proto-object $\mathbf{R}_g$, A_{ct} is the area of the closed contour represented by $\mathbf{F}^{t,\mu}_{ct}$, and θ_d is the angular difference between the principal axis $\theta_{\mathbf{R}_g}$ of the proto-object and the principal axis θ_{ct} of the closed contour represented by $\mathbf{F}^{t,\mu}_{ct}$.

Then the prior shape states are obtained by integrating random factors:

$$\mathbf{X}^m_{\mathbf{R}_g} = \mathbf{X}'_{\mathbf{R}_g} + \mathbf{K}_\omega \boldsymbol{\omega}^m, \tag{5.39}$$

where $\mathbf{K}_\omega$ is a 6×6 diagonal coefficient matrix and $\boldsymbol{\omega}^m$ is a 6×1 random vector whose entries are normally distributed (the mean is 0 and the STD is 1 for each entry).

Step 2: The second step is to calculate predicted contour curves based on the prior shape states. Equation (5.35) can be rewritten as:

$$\mathbf{C} = \begin{pmatrix} \mathbf{B}(z) & 0 \\ 0 & \mathbf{B}(z) \end{pmatrix} (\mathbf{W}\mathbf{X} + \mathbf{F}_{ct}^{t,\mu}) \tag{5.40}$$

where $\mathbf{C} = (x(z), y(z))^T$ is the contour curve parameterized by a real variable z, $\mathbf{B}(z)$ is a $1 \times P$ vector (i.e., the B-Spline basis function vector [150]) whose entries are polynomials in z, and $\mathbf{W}$ is the shape factor calculated as:

$$\mathbf{W} = \begin{pmatrix} 1 & 0 & \mathbf{F}_{ct}^{t,\mu,x} & 0 & 0 & \mathbf{F}_{ct}^{t,\mu,y} \\ 0 & 1 & 0 & \mathbf{F}_{ct}^{t,\mu,y} & \mathbf{F}_{ct}^{t,\mu,x} & 0 \end{pmatrix}. \tag{5.41}$$

Thus, a predicted contour curve $\mathbf{C}_{\mathbf{R}_g}^m$ can be obtained from a prior shape state $\mathbf{X}_{\mathbf{R}_g}^m$ by using (5.40). The number of predicted contour curves is also N_c for each proto-object. It is important to note that a predicted contour curve $\mathbf{C}_{\mathbf{R}_g}^m$ can be seen as one of the candidate representations of the attentional template $\mathbf{F}_{ct}^{t,\mu}$ for the proto-object $\mathbf{R}_g$.

Figure 5.4(b) shows an example of the predicted contour curves for a proto-object.

According to (5.39), the prior probability of a predicted contour curve $\mathbf{C}_{\mathbf{R}_g}^m$ can be calculated as:

$$p_{td}(\mathbf{C}_{\mathbf{R}_g}^m) = \exp[-\frac{1}{2}(\mathbf{X}_{\mathbf{R}_g}^m - \mathbf{X}_{\mathbf{R}_g}')^T(\mathbf{\Sigma}_\omega^m)^{-1}(\mathbf{X}_{\mathbf{R}_g}^m - \mathbf{X}_{\mathbf{R}_g}')], \tag{5.42}$$

where $\mathbf{\Sigma}_\omega^m$ is the covariance matrix of $\boldsymbol{\omega}^m$ in (5.39).

Step 3: The third step is to estimate the observation likelihood $p_{td}(\mathbf{R}_g|\mathbf{C}_{\mathbf{R}_g}^m)$ for each predicted contour curve $\mathbf{C}_{\mathbf{R}_g}^m$. For each predicted contour curve, the model draws several measurement lines, as shown in Figure 5.4(c), which are normal to the predicted contour curve and are spaced equally along the curve. As shown in Figure 5.4(d), the Euclidean distance from the intersection of a measurement line with the predicted curve to the

starting point of the measurement line is called *innovation* and it is written as m_v. The Euclidean distance from a contour feature point (obtained from actual contour $\mathbf{F}_{ct}(l_{wk})$ that is extracted using (4.5)) located on the measurement line to the starting point of the measurement line is called an *observation* and it is written as m_z. Assuming that observations m_z are normally distributed centered on m_v along the measurement line, the observation likelihood of a single measurement line is estimated as:

$$p_{td}(\mathbf{R}_g|m_v, \mathbf{C}^m_{\mathbf{R}_g}) = \frac{1}{N_{cfp}} \sum_{N_{cfp}} \exp(-\frac{|m_z - m_v|^2}{2\sigma_v^2}), \tag{5.43}$$

where N_{cfp} is the number of contour feature points along one measurement line and σ_v is the predefined STD of observations.

Assuming that measurement lines along one predicted curve are independent, the observation likelihood of a predicted contour curve is calculated as:

$$p_{td}(\mathbf{R}_g|\mathbf{C}^m_{\mathbf{R}_g}) = \prod_{N_{ml}} p_{td}(\mathbf{R}_g|m_v, \mathbf{C}^m_{\mathbf{R}_g}), \tag{5.44}$$

where N_{ml} is the number of measurement lines along one predicted curve.

Step 4: The fourth step is to estimate the posterior probability $p_{td}(\mathbf{C}^m_{\mathbf{R}_g}|\mathbf{R}_g)$ of each predicted contour $\mathbf{C}^m_{\mathbf{R}_g}$. It is calculated by using Bayes' theorem:

$$p_{td}(\mathbf{C}^m_{\mathbf{R}_g}|\mathbf{R}_g) = \frac{p_{td}(\mathbf{R}_g|\mathbf{C}^m_{\mathbf{R}_g}) p_{td}(\mathbf{C}^m_{\mathbf{R}_g})}{\sum_{m'=1}^{N_c} p_{td}(\mathbf{R}_g|\mathbf{C}^{m'}_{\mathbf{R}_g}) p_{td}(\mathbf{C}^{m'}_{\mathbf{R}_g})}. \tag{5.45}$$

Step 5: The fifth step is to obtain the estimation of $p_{td}(\mathbf{F}^t_{ct}|\mathbf{R}_g)$ by applying MAP estimation to all predicted contours $\{\mathbf{C}^m_{\mathbf{R}_g}\}$. It can be expressed as:

$$p_{td}(\mathbf{F}^t_{ct}|\mathbf{R}_g) = p_{td}(\mathbf{C}^{m^{max}_g}_{\mathbf{R}_g}|\mathbf{R}_g), \tag{5.46}$$

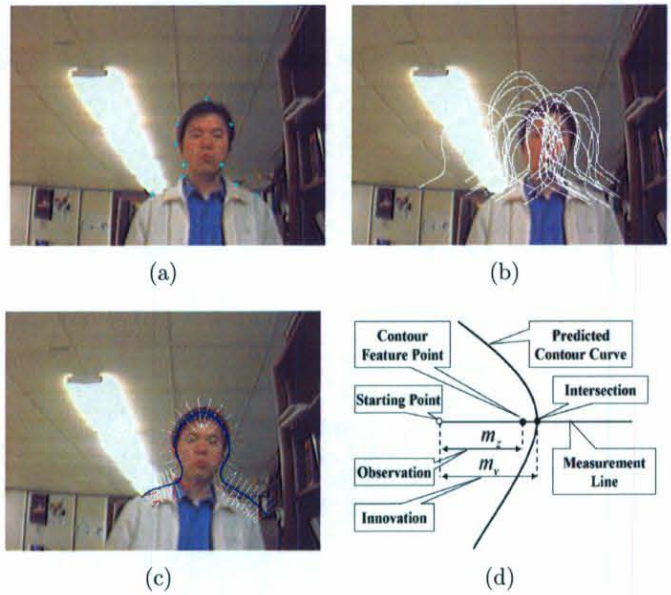


Figure 5.4: Probabilistic top-down biasing techniques in terms of the contour feature. (a) Control points along a contour. (b) Predicted contour curves for a proto-object. (c) Measurement lines along one predicted contour. (d) Innovation and observation along one measurement line.

where

$$m_g^{max} = \arg \max_{m \in \{1,2,\ldots,N_c\}} p_{td}(\mathbf{C}_{\mathbf{R}_g}^m | \mathbf{R}_g). \tag{5.47}$$

***Location-based Top-down Bias in terms of Contour*:** Finally, a location-based top-down bias map in terms of contour is estimated by using (5.34). According to (5.34), the posterior probability of the location $\mathbf{r}_i$ being attended by top-down attention in terms of contour can be estimated as:

$$B_{ct}(\mathbf{r}_i) = p_{td}(\mathbf{r}_i | \mathbf{F}_{ct}^t) = \frac{p_{td}(\mathbf{F}_{ct}^t | \mathbf{r}_i) \times p_{td}(\mathbf{r}_i)}{\sum_{\mathbf{r}_{i'} \in \mathbf{I}} p_{td}(\mathbf{F}_{ct}^t | \mathbf{r}_{i'}) \times p_{td}(\mathbf{r}_{i'})}, \tag{5.48}$$

where B_{ct} denotes the location-based top-down bias in terms of contour.

Biases in terms of Other Feature Dimensions

The observation likelihood $p_{td}(\mathbf{F}^t|\mathbf{r}_i)$ in terms of intensity, red-green pair and blue-yellow pair can be estimated respectively as:

$$p_{td}(\mathbf{F}^t_{int}|\mathbf{r}_i) = \exp(-\frac{1}{2}\frac{|F_{int}(\mathbf{r}_i, l_{wk}) - F^{t,\mu}_{int}|^2}{(F^{t,\sigma}_{int})^2}), \tag{5.49}$$

$$p_{td}(\mathbf{F}^t_{rg}|\mathbf{r}_i) = \exp(-\frac{1}{2}\frac{|F_{rg}(\mathbf{r}_i, l_{wk}) - F^{t,\mu}_{rg}|^2}{(F^{t,\sigma}_{rg})^2}), \tag{5.50}$$

$$p_{td}(\mathbf{F}^t_{by}|\mathbf{r}_i) = \exp(-\frac{1}{2}\frac{|F_{by}(\mathbf{r}_i, l_{wk}) - F^{t,\mu}_{by}|^2}{(F^{t,\sigma}_{by})^2}). \tag{5.51}$$

The observation likelihood $p_{td}(\mathbf{F}^t|\mathbf{r}_i)$ in terms of local orientations can be estimated as:

$$p_{td}(\mathbf{F}^t_{o_\theta}|\mathbf{r}_i) = \begin{cases} F_{o_\theta}(\mathbf{r}_i, l_{wk})/255 & \text{if } \theta = F^t_o \\ 0 & \text{otherwise} \end{cases}, \tag{5.52}$$

where $\theta \in \{0°, 45°, 90°, 135°\}$.

The observation likelihood $p_{td}(\mathbf{F}^t|\mathbf{r}_i)$ in terms of motion can be estimated as:

$$p_{td}(\mathbf{F}^t_{mv}|\mathbf{r}_i) = \begin{cases} F_{mv}(\mathbf{r}_i, l_{wk})/255 & \text{if } F^t_{mv} = 1 \\ 0 & \text{if } F^t_{mv} = 0 \end{cases}. \tag{5.53}$$

Finally, a location-based top-down bias map in terms of the corresponding feature dimension can be obtained by using (5.34). According to (5.34), the posterior probability of the location $\mathbf{r}_i$ being attended by top-down attention in terms of the corresponding feature can be estimated respectively as:

$$B_{int}(\mathbf{r}_i) = p_{td}(\mathbf{r}_i|\mathbf{F}^t_{int}) = \frac{p_{td}(\mathbf{F}^t_{int}|\mathbf{r}_i) \times p_{td}(\mathbf{r}_i)}{\sum_{\mathbf{r}_{i'} \in \mathbf{I}} p_{td}(\mathbf{F}^t_{int}|\mathbf{r}_{i'}) \times p_{td}(\mathbf{r}_{i'})}, \tag{5.54}$$

$$B_{rg}(\mathbf{r}_i) = p_{td}(\mathbf{r}_i|\mathbf{F}^t_{rg}) = \frac{p_{td}(\mathbf{F}^t_{rg}|\mathbf{r}_i) \times p_{td}(\mathbf{r}_i)}{\sum_{\mathbf{r}_{i'} \in \mathbf{I}} p_{td}(\mathbf{F}^t_{rg}|\mathbf{r}_{i'}) \times p_{td}(\mathbf{r}_{i'})}, \tag{5.55}$$

$$B_{by}(\mathbf{r}_i) = p_{td}(\mathbf{r}_i|\mathbf{F}^t_{by}) = \frac{p_{td}(\mathbf{F}^t_{by}|\mathbf{r}_i) \times p_{td}(\mathbf{r}_i)}{\sum_{\mathbf{r}_{i'} \in \mathbf{I}} p_{td}(\mathbf{F}^t_{by}|\mathbf{r}_{i'}) \times p_{td}(\mathbf{r}_{i'})}, \tag{5.56}$$

$$B_{o_\theta}(\mathbf{r}_i) = p_{td}(\mathbf{r}_i|\mathbf{F}^t_{o_\theta}) = \frac{p_{td}(\mathbf{F}^t_{o_\theta}|\mathbf{r}_i) \times p_{td}(\mathbf{r}_i)}{\sum_{\mathbf{r}_{i'} \in \mathbf{I}} p_{td}(\mathbf{F}^t_{o_\theta}|\mathbf{r}_{i'}) \times p_{td}(\mathbf{r}_{i'})}, \tag{5.57}$$

$$B_{mv}(\mathbf{r}_i) = p_{td}(\mathbf{r}_i|\mathbf{F}^t_{mv}) = \frac{p_{td}(\mathbf{F}^t_{mv}|\mathbf{r}_i) \times p_{td}(\mathbf{r}_i)}{\sum_{\mathbf{r}_{i'} \in \mathbf{I}} p_{td}(\mathbf{F}^t_{mv}|\mathbf{r}_{i'}) \times p_{td}(\mathbf{r}_{i'})}, \tag{5.58}$$

where B_{int}, B_{rg}, B_{by}, B_{o_θ}, B_{mv} denote the location-based top-down bias in terms of intensity, red-green pair, blue-yellow pair, orientation in θ, and motion respectively.

5.3.7 Combination of Multi-dimensional Top-down Biases

Since it is possible that multiple task-relevant feature dimensions are used for guiding top-down biasing at a moment, it is required to combine the estimated top-down biases in terms of all task-relevant dimensions. Assuming that top-down biasing in terms of all task-relevant feature dimensions are independent, the observation probability of all task-relevant features can be estimated as:

$$p_{td}(\{\mathbf{F}^t\}|\mathbf{r}_i) = \prod_{f \in \{f_{rel}\}} p_{td}(\mathbf{F}^t_f|\mathbf{r}_i), \tag{5.59}$$

where $p_{td}(\{\mathbf{F}^t\}|\mathbf{r}_i)$ denotes the observation likelihood of a location $\mathbf{r}_i$ being attended by top-down attention in terms of all task-relevant feature dimensions, $\{\mathbf{F}^t\}$ denotes the set of attentional templates in terms of all task-relevant feature dimensions, and $\{f_{rel}\}$ denotes the set of task-relevant feature dimensions.

Finally the total top-down biases can be obtained by using (5.34). According to (5.34), the posterior probability of the location $\mathbf{r}_i$ being attended by top-down attention

in terms of all task-relevant feature dimensions can be estimated as:

$$B_{tot}(\mathbf{r}_i) = p_{td}(\mathbf{r}_i|\{\mathbf{F}^t\}) = \frac{p_{td}(\{\mathbf{F}^t\}|\mathbf{r}_i) \times p_{td}(\mathbf{r}_i)}{\sum_{\mathbf{r}_{i'} \in \mathbf{I}} p_{td}(\{\mathbf{F}^t\}|\mathbf{r}_{i'}) \times p_{td}(\mathbf{r}_{i'})}, \tag{5.60}$$

where B_{tot} denotes the location-based total top-down bias.

5.3.8 Advantages of the Proposed Top-down Biasing Method

Existing methods for top-down attention can be mainly grouped into two categories. The first category is weight-based, such as Navalpakkam's model [39]. This method evaluates a weight for each pre-attentive feature dimension based on the learned representation of the task-relevant object. These weights are used to produce the top-down attention effects by weighting the center-surround difference maps in terms of corresponding pre-attentive feature dimensions during the bottom-up attention process. However, this method might be ineffective in the case that the environment contains distractors which share the relevance with the target in terms of some features, as shown in Figure 2.6 in Chapter 2. That is, the task-relevant object and distractors are possibly both biased by using the similar weights in these shared feature dimensions. The second category is high-level representation based, such as the iconic representation [153, 154]. This method estimates the top-down biases by using a comparison procedure between the entire input image and the representation of the task-relevant object in terms of high-level features. However, one problem of this method is the expensive computational cost since the representation of the task-dependent object is high-level. The other problem of this method is the inflexibility. That is, different high-level representations have to be designed by programmers for a variety of tasks.

Compared with these existing methods, the proposed top-down biasing method has four advantages. The first advantage is effectiveness. It is because the top-down biases are estimated by using both the appearance and salience descriptors of the target. On the one hand, the task-relevant feature(s) are statistically conspicuous compared with a

variety of background distractors presented in the learning process. On the other hand, the statistical appearance descriptors (i.e., attentional template(s)) in terms of the task-relevant feature dimension(s) are used to estimate the top-down biases. As a result, this proposed method can improve the effectiveness in the sense that the task-relevant object can be effectively discriminated from distractors. For example, this proposed top-down biasing method can cope with the case as shown in Figure 2.6 in Chapter 2, in which Navalpakkam's model might fail.

The second advantage is efficiency. The computational complexity of Navalpakkam's Model [39], the iconic representation based method [153, 154] and this proposed method can be approximated as $\mathcal{O}(d_l n)$, $\mathcal{O}(d_h n)$ and $\mathcal{O}(n)$ respectively, where d_l denotes the dimension number of pre-attentive features, d_h denotes the dimension number of a high-level iconic representation of the task-relevant object, and n denotes the number of pixels in an image. Thus, it is obvious that the computation of this proposed method is much cheaper since only one task-relevant feature is used.

The third advantage is adaptability. The task-relevant feature(s) can be autonomously deduced from the learned LTM representation of the task-relevant object such that the requirement of redesigning the representation of the task-relevant object for a variety of tasks is eliminated.

The fourth advantage is robustness. The proposed top-down biasing method gives a bias toward the task-relevant object through a probabilistic procedure by using Bayes' rule and probabilistic estimation techniques. Therefore, the proposed biasing method is robust to work with noise, transformation, occlusion and a variety of viewpoints and illuminative effects. This advantage will be shown in the experimental results in the application of detecting task-relevant objects in Chapter 7 and in the application of target tracking in Chapter 8.

5.4 Combination of Bottom-up Saliency and Top-down Biases

In general, bottom-up attention and top-down attention work together to decide which item is attended. This thesis proposes that the combination of bottom-up saliency and top-down biases is dependent on two successive factors: the conscious factor and the unconscious factor. The conscious factor is the first step and it consciously weights the bottom-up saliency and top-down biases respectively according to the current task, context and learned knowledge. It is called a *conscious weighting step* in this thesis. The unconscious factor is the second step and it automatically combines the weighted bottom-up saliency and weighted top-down biases. It is called an *unconscious combination step* in this thesis.

This thesis proposes a gating mechanism in the conscious weighting step. There are only two cases in any task: 1) only one attention mechanism is used (i.e., bottom-up attention or top-down attention); and 2) both attention mechanisms are simultaneously used. Therefore, only two logic values are used for conscious weighting: 0 and 1. If one attention mechanism is specified by the current task, 1 is set as the weight of that mechanism, i.e., $w_{bu} = 1$ or $w_{td} = 1$. In other words, that attention mechanism is enabled to enter the next unconscious combination step. Otherwise, 0 is set for that mechanism, i.e., $w_{bu} = 0$ or $w_{td} = 0$. In other words, that attention mechanism is inhibited to enter the next unconscious combination step.

The unconscious combination step is difficult due to the multi-modality of bottom-up saliency and top-down biases. However, this thesis proposes a probabilistic method to combine these two modalities at a unified scale based on the fact that the estimated bottom-up saliency and top-down biases are both in probabilistic form. Mathematically, assuming that bottom-up attention and top-down attention are two random events that are independent, the probability of an item being attended can be modeled as the probability of occurrence of either of these two events on that item.

Thus, the total combination process of integrating the conscious and unconscious factors can be expressed in (5.61), which achieves a probabilistic location-based attentional activation map p_{attn}.

$$\begin{cases} p_{attn}(\mathbf{r}_i) = p_{bu}(\mathbf{r}_i) + p_{td}(\mathbf{r}_i|\{\mathbf{F}^t\}) - p_{bu}(\mathbf{r}_i) \times p_{td}(\mathbf{r}_i|\{\mathbf{F}^t\}) & \text{if } w_{bu} = 1 \text{ and } w_{td} = 1 \\ p_{attn}(\mathbf{r}_i) = p_{bu}(\mathbf{r}_i) & \text{if } w_{bu} = 1 \text{ and } w_{td} = 0 \\ p_{attn}(\mathbf{r}_i) = p_{td}(\mathbf{r}_i|\{\mathbf{F}^t\}) & \text{if } w_{bu} = 0 \text{ and } w_{td} = 1. \end{cases} \tag{5.61}$$

The probabilistic location-based attentional activation $p_{attn}(\mathbf{r}_i)$ represents the probability of the location $\mathbf{r}_i$ being attended.

5.5 Estimation of Proto-Object based Attentional Activation

Proto-object based attentional activation represents the probability of a proto-object being attended. It can be estimated in a probabilistic way by a combination of the probability of pixels in a proto-object. Mathematically, that a proto-object is attended can be seen as a random event, denoted as $e_{\mathbf{R}_g}$; and that a location in a proto-object is attended can also be seen as a random event, denoted as $e_{\mathbf{r}_i}$ with $i \in \{1, 2, ..., N_g\}$ where N_g is the number of pixels in the proto-object $\mathbf{R}_g$. According to Duncan's IC hypothesis [49], i.e., a competitive advantage over the whole object is produced by directing attention to a spatial location in that object, the probability of $e_{\mathbf{R}_g}$ can be calculated as:

$$p(e_{\mathbf{R}_g}) = \frac{1}{N_g} p(e_{\mathbf{r}_1} \vee e_{\mathbf{r}_2} \vee ... \vee e_{\mathbf{r}_{N_g}}), \tag{5.62}$$

where $\vee$ denotes "logic or" operator, and $\frac{1}{N_g}$ is included in order to eliminate the influence of the proto-object's size.

Based on the space-based attention theory, only one location can be attended at a

moment. It is thereby reasonable to assume that all events $e_{\mathbf{r}_i}$ are mutually exclusive. The probability of a proto-object being attended, denoted as $p_{attn}(\mathbf{R}_g)$, can be calculated by extending (5.62):

$$p_{attn}(\mathbf{R}_g) = \frac{1}{N_g} \sum_{\mathbf{r}_i \in \mathbf{R}_g} p_{attn}(\mathbf{r}_i). \tag{5.63}$$

The proto-object based attentional activation map is composed of the probability $p_{attn}(\mathbf{R}_g)$ of each proto-object. The focus of attention is directed to the proto-object with maximal proto-object based attentional activation. The dynamical shift of the focus of attention is produced by allowing the next most active proto-object to subsequently become the winner.

This estimation method for proto-object based attentional activation is also consistent with Sun's model [42]. The attentional activation of a proto-object is modeled as a combination of all activation contributions coming from the pixels within that proto-object in the sense that those pixels work together to compete with their common competitors and cooperate with each other.

It can be seen that a location-based attentional activation map (i.e., (5.61)) and a proto-object based attentional activation map (i.e. (5.63)) are both estimated. It indicates that space-based attention and object-based attention are modeled into a unified framework in this proposed cognitive perception paradigm.

5.6 Conclusion

This chapter has presented the attentional selection stage in the proposed cognitive visual perception paradigm. Four modules have been presented in the attentional selection stage: bottom-up competition, top-down biasing, combination of bottom-up saliency and top-down biases, and estimation of proto-object based attentional activation. The bottom-up competition module models the bottom-up attention mechanism and yields a probabilistic location-based bottom-up saliency map. The top-down biasing module models the top-down attention mechanism and yields a probabilistic location-based top-

down bias map. The combination module combines the saliency map and bias map to yield a probabilistic location-based attentional activation map. Finally a proto-object based attentional activation map is achieved and it is used to guide the focus of attention.

There are several advantages in the proposed attentional selection stage.

1. The most important advantage of the proposed attentional selection stage is the novel top-down biasing method proposed based on Duncan's IC hypothesis [49]. This method uses one or a few conspicuous low-level task-relevant feature(s) of the task-relevant object to guide top-down attention. The advantages of this method include effectiveness, efficiency, adaptability and robustness.

2. A bottom-up competition method is proposed by extending Itti's bottom-up attention model [38]. There are two advantages in this new method: 1) Contour and motion features are included in the bottom-up competition such that conspicuousness in terms of contour and motion can be achieved; 2) A probabilistic bottom-up saliency map is estimated, with the result that combination of bottom-up saliency and top-down biases can be performed at a unified probabilistic scale.

3. A method is proposed for combining bottom-up saliency and top-down biases by integrating a conscious gating factor and an unconscious combination factor.

4. A probabilistic method is proposed to estimate the proto-object based attentional activation.

Chapter 6

Post-attentive Perception

6.1 Introduction

Once a proto-object is selected to be attended, it is sent into the post-attentive perception stage for high-level perceptual analysis. Although the post-attentive perception stage could involve a variety of processing according to the tasks, this thesis asserts that the main objective of the post-attentive perception stage is to interpret the attended object in more detail. Detailed interpretation aims to produce an appropriate action at the current moment, to correctly update the corresponding LTM object representation at the current moment and to consciously guide the top-down attention at the next moment.

Four functional modules are included in the post-attentive perception stage. The first module is perceptual completion processing. An object is always composed of several parts. According to Duncan's IC hypothesis [49], other parts of an object presented in the current scene are also attended once one part (i.e., the attended proto-object) of that object is selected by attention. This indicates that perceptual completion processing is required to perceive the complete region of the attended object post-attentively. This thesis uses the term *attended object* to represent one or all of the proto-objects in the complete region being attended.

The second module is the extraction of post-attentive features. Post-attentive features

are a type of high-level features estimated based on the pre-attentive features in the attended object. In order to interpret the attended object in more detail, these extracted post-attentive features are used to build a high-level representation of the attended object in WM. This high-level representation is termed as *WM object representation* in this thesis, in order to distinguish it from the *LTM object representation* that is the learned object representation stored in LTM.

The third module is object recognition. The WM representation of the attended object is used for recognizing it. The object recognition module functions as a decision unit that determines to which LTM object representation and/or to which instance of that representation the attended object belongs. This module is used in three procedures, including perceptual completion processing, unsupervised learning of the attended object and action selection. It is important to note that the term *object* used in this chapter can represent either an entire object or a proto-object.

The fourth module is the development of LTM object representations. The term *development* indicates two types of operations. The first operation is constructing a structure of LTM object representations. The second operation is dynamically learning the corresponding LTM object representation given the attended object at each moment. The corresponding LTM object representation can be retrieved using the object recognition module. The WM representation of the attended object can be regarded as a training sample for learning. As a result, the learned LTM object representation can incorporate all instances of that object during the lifelong cognitive development process of the robots. The developed LTM object representations can be used for top-down biasing, perceptual completion processing, object recognition and action selection at the next moments.

These four modules indicate that all the work in the post-attentive perception stage is centered around the attended object so that the attended object can be completely perceived post-attentively.

In fact, these four modules are interactive during the post-attentive perception stage.

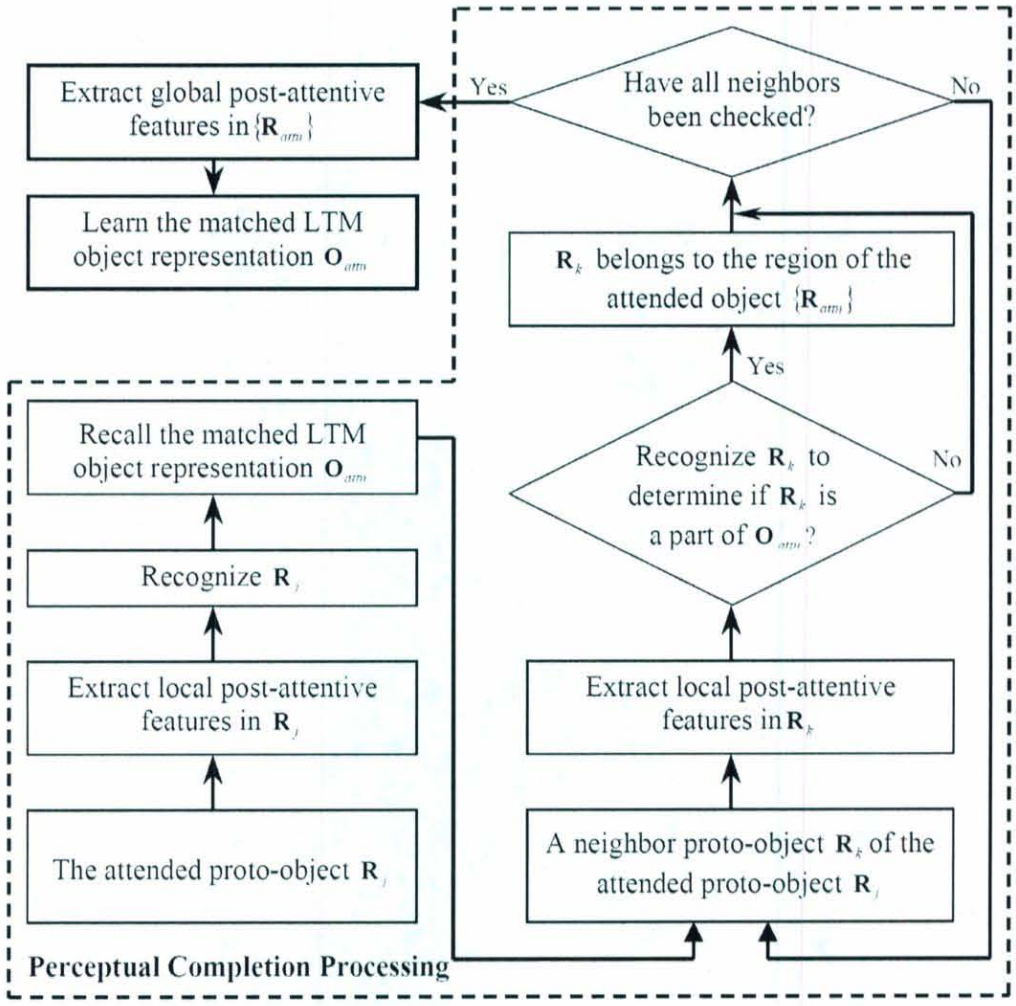


Figure 6.1: The flowchart of the post-attentive perception stage. Four modules, including perceptual completion processing, extraction of post-attentive features, object recognition and development of LTM object representations, are interactive in the post-attentive stage.

In particular, extraction of post-attentive features and recognition of attended proto-objects are both performed during the procedure of perceptual completion processing. The flow chart of the post-attentive perception can be illustrated in Figure 6.1. The following sections will give the detailed explanation of this flow chart.

Although post-attentive perception can perform at any spatial scale, the proposed perception paradigm asserts that the working scale l_{wk} is more suitable for post-attentive

perception. It is due to two facts. The first fact is that attentional selection performs at the working scale and the second fact is that some results of post-attentive perception at the current moment are then used in the attentional selection at the next moments.

This chapter is organized as follows. Section 6.2 presents the perceptual completion processing. Section 6.3 presents the extraction of post-attentive features. Since the object recognition algorithms are based on LTM object representations, section 6.4 presents the development of LTM object representations and then section 6.5 presents the object recognition.

6.2 Perceptual Completion Processing

The module of perceptual completion processing works around the attended proto-object, denoted as $\mathbf{R}_j$, to achieve the complete object region. It consists of two steps.

The first step is recognition of the attended proto-object. This step explores LTM object representations in order to determine to which LTM object representation the attended proto-object belongs. The post-attentive features in the attended proto-object are used for recognition. The extraction of post-attentive features will be presented in section 6.3. The recognition algorithm will be presented in section 6.5. After recognition, the matched LTM object representation, denoted as $\mathbf{O}_{attn}$, is recalled from LTM for the following completion processing. The routine of this step is shown in the lower left side of Figure 6.1.

The second step is completion processing. The purpose of this step is to determine which neighbor proto-objects around $\mathbf{R}_j$ belong to the compete region of the attended object based on $\mathbf{O}_{attn}$. The routine of this step is given as follows.

1. If the local coding of $\mathbf{O}_{attn}$ includes multiple parts, several candidate proto-objects, which are spatially close to $\mathbf{R}_j$, are selected from the current scene. They are termed as *neighbors* and denoted as a set $\{\mathbf{R}_k\}$.

2. The local post-attentive features are extracted in one neighbor $\mathbf{R}_k$.

3. The neighbor $\mathbf{R}_k$ is recognized using the local post-attentive features of the neighbor and the matched LTM object representation $\mathbf{O}_{attn}$. If the neighbor is recognized as a part of $\mathbf{O}_{attn}$, it will be labeled as a part of the attended object. Otherwise, it will be eliminated.

4. Continue *item 2* and *item 3* iteratively until all neighbors have been checked.

These labeled proto-objects constitute the complete region of the attended object, which is denoted as a set $\{\mathbf{R}_{attn}\}$ in the following text.

6.3 Extraction of Post-attentive Features

6.3.1 Definition of Post-attentive Features

Post-attentive features can be defined as a type of distinct high-level features that has the capability to effectively represent the attended object in the cognitive perception process. They are denoted as $\tilde{\mathbf{F}}$.

Although a variety of methods have been proposed to extract high-level features for distinct tasks, such as SIFT feature [7] for object recognition, post-attentive features should be estimated using a distinct form in order to satisfy the special requirement of cognitive visual perception. The function of post-attentive features can be presented as follows. The extracted post-attentive features are first used to form a WM representation of the attended object. The WM representation is then used to recognize the attended object and train the corresponding LTM object representation. Finally, the learned LTM object representation is used to guide action selection at the current moment and it is also used to guide top-down biasing at the next moment. Therefore, this thesis proposes two rules for extracting post-attentive features.

The first rule is that post-attentive features are estimated by using statistics in terms of pre-attentive feature dimensions within the attended object. The use of statistics is based on the fact that the perceived data can be reduced to more manageable amounts

by using the statistical structure in the data to recode the information it contains [155]. This indicates that the WM representation of an object can be estimated by using the statistics of the contributions coming from the pixels in that object. The advantage of keeping the same types of feature dimensions with pre-attentive features is to conveniently guide top-down biasing for the next moment.

The second rule is that post-attentive features should be estimated not only using the appearance values but also using the saliency values that have been obtained from the bottom-up competition module. Since the saliency values represent the conspicuousness of an object compared with other objects in terms of a feature dimension, the inclusion of them is helpful to guide top-down biasing at the next moment.

In order to satisfy these two rules, the post-attentive features are estimated based on two facts. The first fact is that pre-attentive features can be mainly grouped into two categories. The first category is the global feature, including the contour $\mathbf{F}_{ct}$. The second category is the local features, including intensity, red-green pair, blue-yellow pair and orientation energy. The second fact is that the top-down biasing module requires both appearance and salience values of the task-relevant object for estimating biases. Therefore, the post-attentive features $\tilde{\mathbf{F}}$ consist of global features $\tilde{\mathbf{F}}_{gb}$ and local features $\tilde{\mathbf{F}}_{lc}$. Each $\tilde{\mathbf{F}}$ also consists of appearance component $\tilde{\mathbf{F}}^{a}$ and salience component $\tilde{\mathbf{F}}^{s}$.

6.3.2 Local Post-attentive Features

Structure of Local Post-attentive Features

Local post-attentive features $\tilde{\mathbf{F}}_{lc}$ can be defined as a type of high-level representations of all local parts of an attended object. Each proto-object, denoted as $\mathbf{R}_{j}^{attn}$, in the complete region being attended (i.e., $\mathbf{R}_{j}^{attn} \in \{\mathbf{R}_{attn}\}$) is thereby the unit for estimating local post-attentive features.

Correspondingly, local post-attentive features can be estimated as a set, each entry of which represents the statistical properties in terms of appearance and salience of a $\mathbf{R}_{j}^{attn}$.

Thus, the set of local post-attentive features can be expressed as:

$$\{\tilde{\mathbf{F}}_{lc}\} = \{\tilde{\mathbf{F}}_{lc}(\mathbf{R}_j^{attn})\}_{\forall \mathbf{R}_j^{attn} \in \{\mathbf{R}_{attn}\}}. \tag{6.1}$$

Each entry of the post-attentive feature includes the appearance component $\tilde{\mathbf{F}}_{lc}^{a}(\mathbf{R}_j^{attn})$ and salience component $\tilde{\mathbf{F}}_{lc}^{s}(\mathbf{R}_j^{attn})$, as shown in the following:

$$\tilde{\mathbf{F}}_{lc}(\mathbf{R}_j^{attn}) = \begin{pmatrix} \tilde{\mathbf{F}}_{lc}^{a}(\mathbf{R}_j^{attn}) & \tilde{\mathbf{F}}_{lc}^{s}(\mathbf{R}_j^{attn}) \end{pmatrix}^T. \tag{6.2}$$

Since local features include intensity, red-green pair, blue-yellow pair and local orientations, the appearance component and salience component of an entry can be expressed respectively as:

$$\tilde{\mathbf{F}}_{lc}^{a}(\mathbf{R}_j^{attn}) = \begin{pmatrix} \tilde{\mathbf{F}}_{int}^{a}(\mathbf{R}_j^{attn}) & \tilde{\mathbf{F}}_{rg}^{a}(\mathbf{R}_j^{attn}) & \tilde{\mathbf{F}}_{by}^{a}(\mathbf{R}_j^{attn}) & \tilde{\mathbf{F}}_{o}^{a}(\mathbf{R}_j^{attn}) \end{pmatrix}^T, \tag{6.3}$$

$$\tilde{\mathbf{F}}_{lc}^{s}(\mathbf{R}_j^{attn}) = \begin{pmatrix} \tilde{F}_{int}^{s}(\mathbf{R}_j^{attn}) \\ \tilde{F}_{rg}^{s}(\mathbf{R}_j^{attn}) \\ \tilde{F}_{by}^{s}(\mathbf{R}_j^{attn}) \\ \tilde{F}_{o_{0^\circ}}^{s}(\mathbf{R}_j^{attn}) \\ \tilde{F}_{o_{45^\circ}}^{s}(\mathbf{R}_j^{attn}) \\ \tilde{F}_{o_{90^\circ}}^{s}(\mathbf{R}_j^{attn}) \\ \tilde{F}_{o_{135^\circ}}^{s}(\mathbf{R}_j^{attn}) \end{pmatrix}. \tag{6.4}$$

As shown in the module of perceptual completion processing (i.e., section 6.2), each entry $\tilde{\mathbf{F}}_{lc}(\mathbf{R}_j^{attn})$ is estimated during the corresponding loop in the second step of perceptual completion processing. Estimation of appearance components and salience components is given in the following subsections.

Appearance Components in terms of Intensity and Colors

The appearance components in terms of intensity, red-green pair and blue-yellow pair are estimated by using two types of statistical measures. One is low-order statistical measures, including the mean of a $\mathbf{R}_j^{attn}$ in terms of the corresponding feature dimension. The other is high-order statistical measures using histograms, i.e., the histogram of a $\mathbf{R}_j^{attn}$ in terms of corresponding feature dimension. The low-order statistical measures are used as types of low-level appearance representations of a $\mathbf{R}_j^{attn}$. Since these low-level representations have low computational cost, they are used as task-relevant features in the attentional selection stage to guide top-down biasing. The high-order statistical measures are used as types of high-level appearance representations of a $\mathbf{R}_j^{attn}$. Since these high-level representations have high computational cost, they are used in the post-attentive perception stage for precise analysis, such as object recognition and development of LTM object representations. Thus, the appearance components in terms of intensity, red-green pair and blue-yellow can be expressed respectively as:

$$\tilde{\mathbf{F}}_{int}^{a}(\mathbf{R}_j^{attn}) = \begin{pmatrix} \tilde{\mu}_{int}^{a,j} & \mathbf{H}_{int}^{a,j} \end{pmatrix}^T, \tag{6.5}$$

$$\tilde{\mathbf{F}}_{rg}^{a}(\mathbf{R}_j^{attn}) = \begin{pmatrix} \tilde{\mu}_{rg}^{a,j} & \mathbf{H}_{rg}^{a,j} \end{pmatrix}^T, \tag{6.6}$$

$$\tilde{\mathbf{F}}_{by}^{a}(\mathbf{R}_j^{attn}) = \begin{pmatrix} \tilde{\mu}_{by}^{a,j} & \mathbf{H}_{by}^{a,j} \end{pmatrix}^T, \tag{6.7}$$

where $\tilde{\mu}^{a,j}$ denotes the appearance mean in terms of a feature dimension of a $\mathbf{R}_j^{attn}$, $\mathbf{H}^{a,j}$ denotes the appearance histogram in terms of a feature dimension of the $\mathbf{R}_j^{attn}$.

The histograms with fixed bin size are used to estimate $\mathbf{H}_{int}^{a,j}$, $\mathbf{H}_{rg}^{a,j}$ and $\mathbf{H}_{by}^{a,j}$. The intensity histogram $\mathbf{H}_{int}^{a,j}$ has 10 bins. The red-green histogram $\mathbf{H}_{rg}^{a,j}$ and blue-yellow histogram $\mathbf{H}_{by}^{a,j}$ have 20 bins respectively. The procedure for calculating $\mathbf{H}_{int}^{a,j}$, $\mathbf{H}_{rg}^{a,j}$ and $\mathbf{H}_{by}^{a,j}$ can be expressed as follows. For each feature dimension $f \in \{int, rg, by\}$, each pixel

$\mathbf{r}_i$ in $\mathbf{R}_j^{attn}$ is accumulated into the corresponding bin in the histogram of that feature dimension f according to its pre-attentive feature value $F_f(\mathbf{r}_i, l_{wk})$.

It is important to note that the above settings about the bin sizes of $\mathbf{H}_{int}^{a,j}$, $\mathbf{H}_{rg}^{a,j}$ and $\mathbf{H}_{by}^{a,j}$ are obtained empirically. Furthermore, our experiments show that the variation of the current settings can also work.

Appearance Component in terms of Local Orientations

As shown in section 5.3.5 in Chapter 5, the attentional template is directly built by using the orientation direction when the local orientation is selected as the task-relevant feature during the top-down biasing procedure. Thus, only high-order statistical measures are used to estimate the appearance component in terms of local orientations.

In order to build a rotation-invariant LTM object representation, i.e., the representation is robust to the object's rotation, the appearance component in terms of local orientations of a $\mathbf{R}_j^{attn}$ is estimated with respect to the principal axis of $\mathbf{R}_j^{attn}$. Calculation of the principal axis of a proto-object has been given in (4.25) in Chapter 4.

Pixels with Available Orientation: An issue about local orientations should be discussed first. It is obvious that the orientation of a pixel is unavailable when that pixel does not have a large change rate of intensity, i.e. when the orientation energy is small at that pixel. Therefore, only the pixels whose orientation can be obtained are used to build the appearance component. These pixels are termed as *available pixels*. The set of these pixels in a $\mathbf{R}_j^{attn}$ can be obtained by using (6.8):

$$\begin{cases} \mathbf{r}_i \in \{\mathbf{r}_a^j\} & \text{if } F_{ct}(\mathbf{r}_i, l_{wk}) \geq \tau_a \text{ and } \mathbf{r}_i \in \mathbf{R}_j^{attn} \\ \mathbf{r}_i \notin \{\mathbf{r}_a^j\} & \text{otherwise} \end{cases}, \tag{6.8}$$

where $\{\mathbf{r}_a^j\}$ denotes the set of available pixels in a $\mathbf{R}_j^{attn}$, and τ_a denotes the threshold used to determine if the orientation energy at a pixel is large enough. The pre-attentive feature

$F_{ct}(\mathbf{r}_i, l_{wk})$ is used here based on the fact that it can represent the maximal orientation energy among all orientations at a pixel $\mathbf{r}_i$ at the working scale l_{wk}, which can be seen clearly in (4.5) in Chapter 4.

Relative Orientation: Given the principal axis of a $\mathbf{R}_j^{attn}$, the relative orientation of an available pixel $\mathbf{r}_a^j$ with respect to the principal axis of $\mathbf{R}_j^{attn}$ can be estimated. It is termed as *relative orientation* in this thesis. Since pre-attentive segmentation performs at the working scale, the relative orientation feature is also estimated at the working scale.

The calculation procedure of the relative orientation consists of two steps. The first step is to calculate the absolute orientation of each available pixel in a $\mathbf{R}_j^{attn}$ at the working scale. It is implemented based on the fact that the a pixel has the maximal orientation energy in its absolute orientation and it can be expressed as:

$$F_{ao}(\mathbf{r}_a^j) = \arg \max_{\theta \in \{0°, 45°, 90°, 135°\}} F_{o\theta}(\mathbf{r}_a^j, l_{wk}), \tag{6.9}$$

where $F_{ao}(\mathbf{r}_a^j)$ denotes the absolute orientation of an available pixel $\mathbf{r}_a^j$ in $\mathbf{R}_j^{attn}$ at the working scale.

The second step is to calculate the relative orientation of each available pixel in a $\mathbf{R}_j^{attn}$ at the working scale. It can be expressed as:

$$F'_{ro}(\mathbf{r}_a^j) = F_{ao}(\mathbf{r}_a^j) - \theta_{\mathbf{R}_j^{attn}}, \tag{6.10}$$

where $\theta_{\mathbf{R}_j^{attn}}$ represents the principal axis of $\mathbf{R}_j^{attn}$.

Since only four preferred orientations are used in the proposed perception paradigm, the obtained relative orientation $F'_{ro}(\mathbf{r}_a^j)$ is then categorized into a direction of $\{0°, 45°, 90°, 135°\}$, achieving the final relative orientation, denoted as $F_{ro}(\mathbf{r}_a^j)$.

***Appearance Histogram in terms of Local Orientations*:** A histogram, denoted as $\mathbf{H}_o^{a,j}$, is used to represent the appearance component in terms of local orientations. This histogram has 4 bins and each bin corresponds to an orientation $\theta \in \{0°, 45°, 90°, 135°\}$.

The estimation procedure of $\mathbf{H}_o^{a,j}$ can be expressed as follows. Each available pixel $\mathbf{r}_a^j$ in a $\mathbf{R}_j^{attn}$ is accumulated into the corresponding bin in the histogram of local orientations according to its relative orientation $F_{ro}(\mathbf{r}_a^j)$.

Finally, the appearance component in terms of local orientations can be expressed as:

$$\tilde{\mathbf{F}}_o^a(\mathbf{R}_j^{attn}) = \mathbf{H}_o^{a,j}. \tag{6.11}$$

Salience Components

Since saliency values represent the conspicuousness of a proto-object compared with other objects, the salience components of local post-attentive features can be estimated using the low-order statistical measure, i.e., the mean of conspicuity of the attended proto-object. They can be expressed as:

$$\tilde{F}_{int}^s(\mathbf{R}_j^{attn}) = \tilde{\mu}_{int}^{s,j}, \tag{6.12}$$

$$\tilde{F}_{rg}^s(\mathbf{R}_j^{attn}) = \tilde{\mu}_{rg}^{s,j}, \tag{6.13}$$

$$\tilde{F}_{by}^s(\mathbf{R}_j^{attn}) = \tilde{\mu}_{by}^{s,j}, \tag{6.14}$$

$$\tilde{F}_{o_\theta}^s(\mathbf{R}_j^{attn}) = \tilde{\mu}_{o_\theta}^{s,j}, \tag{6.15}$$

where $\tilde{\mu}^{s,j}$ denotes the mean of conspicuity in terms of a feature dimension in a $\mathbf{R}_j^{attn}$, the conspicuity values have been obtained respectively using (5.6), (5.7), (5.8) and (5.9) in Chapter 5, and $\theta \in \{0°, 45°, 90°, 135°\}$.

6.3.3 Global Post-attentive Feature

As shown in Figure 6.1, the global post-attentive feature is estimated after the complete region of the attended object, i.e., $\{\mathbf{R}_{attn}\}$, is obtained. The contour is used as the global feature. The pre-attentive feature in terms of contour, i.e.,$\mathbf{F}_{ct}$, is obtained from orientation energy, as shown in (4.5). However, $\mathbf{F}_{ct}$ does not provide a concrete representation of a contour. At present, various representations of a contour have been proposed, such as [156], in which a set of points along a contour are extracted by satisfying the condition that any two neighbor points have a fixed distance. Choosing a representation of the contour used for the global post-attentive feature is dependent on the balance between precision and computational cost. In other words, the number of points included in the representation should be as small as possible, whereas the amount of information contained in those points should be enough to describe the contour accurately. Thus, this thesis uses control points to represent a contour in cooperation with the B-Spline technique [150]. This representation has been shown in (5.35) in Chapter 5. Thus, the estimation of global post-attentive features $\tilde{\mathbf{F}}_{gb}$ includes two steps. The first step is to extract control points of the attended object's contour. The second step is to estimate the appearance and salience components at these extracted control points.

Extraction of Control Points

Control points can be defined as locations with significant changes in curvature. Physiological research has shown that hypercomplex cells in the visual cortex are responsible for localizing these locations based on the fact that excitatory influences from the small receptive field and inhibitory influences from the large receptive field converge in the hypercomplex cell [82]. The detailed mechanism of hypercomplex cells has been presented in section 2.2.5 in Chapter 2. This thesis therefore proposes a two-stage process to extract control points. The first stage is across-scale interaction by taking the difference of the responses of Gabor filters at the same central positions and orientations, but of different receptive sizes (i.e., at different scales). The second stage is across-orientation

cooperation based on the fact that the location of a control point has the maximal difference of responses over its neighbors in at least two orientation directions. An advantage of this method is that it makes use of already available orientation energy features. This method is expressed from (6.16) to (6.18). Equation (6.16) calculates the across-scale difference in order to implement the first stage. Equation (6.17) implements the second stage. It first determines whether a point has the maximal across-scale difference in an orientation and then accumulates the number of satisfied orientations. Finally, equation (6.18) determines if a point is a control point by checking its accumulated orientation number:

$$F'_{o_\theta}(\mathbf{r}_i, l_c, l_s) = |F_{o_\theta}(\mathbf{r}_i, l_c) \ominus F_{o_\theta}(\mathbf{r}_i, l_s)|, \tag{6.16}$$

$$d(\mathbf{r}_i) = \dim\left(\{\theta\} : F'_{o_\theta}(\mathbf{r}_i, l_c, l_s) = \max_{\mathbf{r}\in\mathbb{N}_{\mathbf{r}_i}} F'_{o_\theta}(\mathbf{r}, l_c, l_s)\right), \tag{6.17}$$

$$\begin{cases} \mathbf{r}_i \in \{\mathbf{r}_{cp}\} & \text{if } d(\mathbf{r}_i) \geq 2 \\ \mathbf{r}_i \notin \{\mathbf{r}_{cp}\} & \text{otherwise} \end{cases}, \tag{6.18}$$

where scale $l_c = 2$, scale $l_s = 5$, $\ominus$ is across-scale subtraction, $\mathbb{N}_{\mathbf{r}_i}$ are neighbors of a pixel $\mathbf{r}_i$, the set $\{\theta\}$ includes θ values that satisfy the condition, $\theta \in \{0°, 45°, 90°, 135°\}$, function $\dim(\cdot)$ is to obtain the entry number of the set $\{\theta\}$, and $\{\mathbf{r}_{cp}\}$ denotes the set of control points.

Figure (6.2) shows some examples of the extracted control points.

The extracted control points at other views are transformed into the reference frame by using affine transformation, whose parameters are estimated by matching SIFT keypoints between the image at the present view and the reference image frame.

To extract global control points, a band range $\mathbf{R}_c$ is set manually along the object's global contour in the reference frame to filter out local control points.

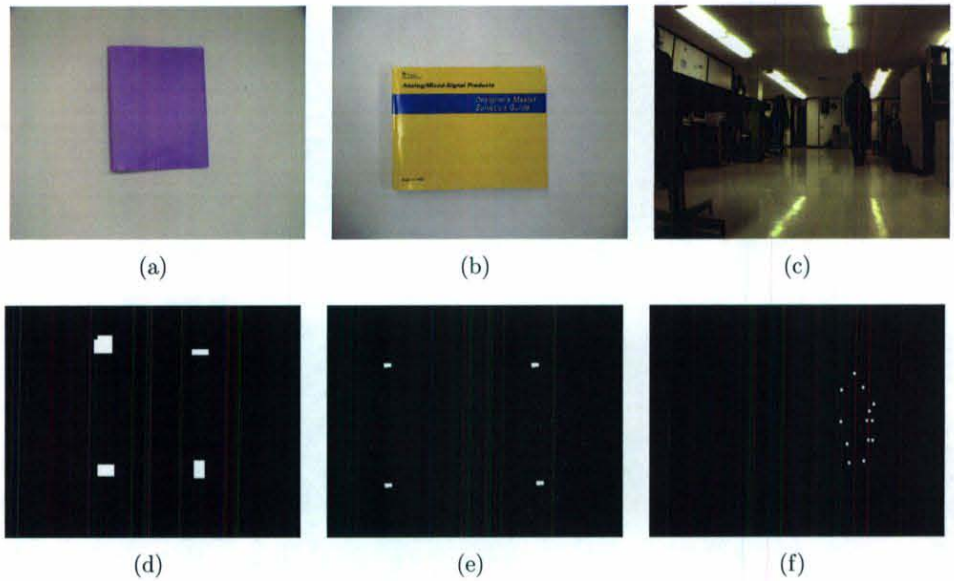


Figure 6.2: Extraction of global control points in the post-attentive perception stage. (a)-(c) Original images. (d)-(f) Global control points extracted.

Estimation of Appearance and Salience Components

The global post-attentive feature is finally estimated as a set, each entry of which represents the statistics in terms of appearance and salience at a control point $\mathbf{r}_{cp}$. Thus, the set of global post-attentive features can be expressed as:

$$\{\tilde{\mathbf{F}}_{gb}\} = \{\tilde{\mathbf{F}}_{gb}(\mathbf{r}_{cp})\}_{\forall \mathbf{r}_{cp}}. \tag{6.19}$$

Each entry of the global post-attentive feature set includes the appearance component $\tilde{\mathbf{F}}^{a}_{gb}(\mathbf{r}_{cp})$ and salience component $\tilde{\mathbf{F}}^{s}_{gb}(\mathbf{r}_{cp})$, as shown in the following:

$$\tilde{\mathbf{F}}_{gb}(\mathbf{r}_{cp}) = \begin{pmatrix} \tilde{\mathbf{F}}^{a}_{gb}(\mathbf{r}_{cp}) & \tilde{\mathbf{F}}^{s}_{gb}(\mathbf{r}_{cp}) \end{pmatrix}^{T}. \tag{6.20}$$

The appearance component of an entry consists of spatial coordinates, denoted as

$(x_{\mathbf{r}_{cp}}, y_{\mathbf{r}_{cp}})$, in the reference frame at a global control point $\mathbf{r}_{cp}$. It can be expressed as:

$$\tilde{\mathbf{F}}^{a}_{gb} = \begin{pmatrix} x_{\mathbf{r}_{cp}} & y_{\mathbf{r}_{cp}} \end{pmatrix}^{T}. \tag{6.21}$$

The salience component of an entry is built using the conspicuity value $F^{s}_{ct}(\mathbf{r}_{cp})$ in terms of pre-attentive contour feature at a global control point $\mathbf{r}_{cp}$, which is calculated using (5.11). It can be expressed as:

$$\tilde{\mathbf{F}}^{s}_{gb}(\mathbf{r}_{cp}) = F^{s}_{ct}(\mathbf{r}_{cp}). \tag{6.22}$$

6.4 Development of LTM Object Representations

6.4.1 Functions of LTM Object Representations

According to object-based visual attention theory [25] and the object-based visual perception idea [36], perception and action are both performed using the fundamental unit of objects. This indicates that LTM object representations can be seen as the mental carriers of knowledge.

LTM object representations mainly have two functions in the robotic perception-action loop. The first function is to guide top-down biasing during the process of perception. Given the task-relevant object, its LTM representation is recalled to deduce the task-relevant feature(s), which then constitute the attentional template(s) in WM to estimate the top-down biases.

The second function is to guide action selection during the process of action. LTM object representations can be used to encode cognitive perception-action mapping. This mapping represents the association between the attentional states and candidate actions. Since attentional selection is object-based in the proposed paradigm, the perception-action mapping actually represents the correspondence between the learned LTM object representations and candidate actions. This cognitive perception-action decision pro-

cess can be described as follows. Once an object is attended at a moment, it is recognized by exploring the existing LTM object representations, and then the matched instance of an LTM object representation leads to an appropriate action according to the perception-action mapping. It can be seen that two procedures are required in this cognitive perception-action decision process. The first procedure is the recognition of the attended object, which will be presented in this chapter. The second procedure is the learning of the mapping between attentional states and actions, which is not addressed in this thesis.

Based on the above discussion, the main objective of developing an LTM object representation is to encode various instances of the object into a representation so that it can be effectively used to fulfill the above two functions.

The following subsections are organized as follows. Subsection 6.4.2 to subsection 6.4.6 present the structure of the proposed LTM object representations. Subsection 6.4.7 presents the algorithm for learning the LTM object representations.

6.4.2 Neural Mechanisms for Object Codings

Neuropsychological studies have revealed the existence of two parallel forms of object codings in space [157]. One is within-object coding, in which elements are coded as local parts of an object. It represents local properties of an object, including local part formation and local features. The other is between-object coding, in which elements are coded as independent objects. This indicates that global attributes of an object are also encoded into its LTM representation.

6.4.3 Infrastructure of LTM Object Representations

Accordingly, this thesis proposes a dual-coding LTM object representation $\mathbf{O}$ that includes global coding $\mathbf{O}_{gb}$ and local coding $\mathbf{O}_{lc}$. Since contour is composed of elements which characterize an entire object, it is used to build global coding. Intensity, red-green pair, blue-yellow pair and local orientations are used to build local coding.

According to the structure of post-attentive features, the proposed LTM object representation also includes two descriptors: appearance $\mathbf{O}^a$ and salience $\mathbf{O}^s$. The appearance descriptor represents the appearance value of each feature dimension. The salience descriptor represents conspicuousness of each feature dimension and it is used to deduce the task-relevant feature(s) at the beginning of top-down biasing.

Thus, the infrastructure of the proposed LTM object representation can be expressed as:

$$\mathbf{O} = \begin{pmatrix} \mathbf{O}_{gb} & \mathbf{O}_{lc} \end{pmatrix} = \begin{pmatrix} \mathbf{O}^a_{gb} & \mathbf{O}^a_{lc} \\ \mathbf{O}^s_{gb} & \mathbf{O}^s_{lc} \end{pmatrix}, \tag{6.23}$$

where $\mathbf{O}_{gb}$ denotes global coding and $\mathbf{O}_{lc}$ denotes local coding.

6.4.4 PNN based LTM Object Representations

PNNs and Extended PNNs

The probabilistic neural network (PNN) [53] is used in this thesis to construct LTM object representations. A PNN consists of an *input layer*, one or several *hidden layers* and an *output layer*. As an example, a three-layer PNN has been shown in Figure 6.3. The input layer receives the input signals. The first hidden layer is composed of radial basis functions (RBF) and Gaussian distributions are always used as RBFs. Each parent node in an upper hidden layer or an output layer is a probabilistic mixture of its son nodes in its lower hidden layer. The term *probabilistic mixture* denotes a weighted sum of a set of probabilistic distributions, e.g., a mixture of a set of Gaussian distributions. Thus, the nodes in higher hidden layers and the output layer are multi-modal distributions.

Based on this multi-layer structure, a PNN can encode an object as a hierarchical mixture distribution. In other words, a PNN can hierarchically and probabilistically incorporate various instances obtained under different viewing conditions. Thus, the advantage of using PNNs for constructing LTM object representations is the robustness to perceptual uncertainties, such as noise, changes of lighting conditions and changes of views.

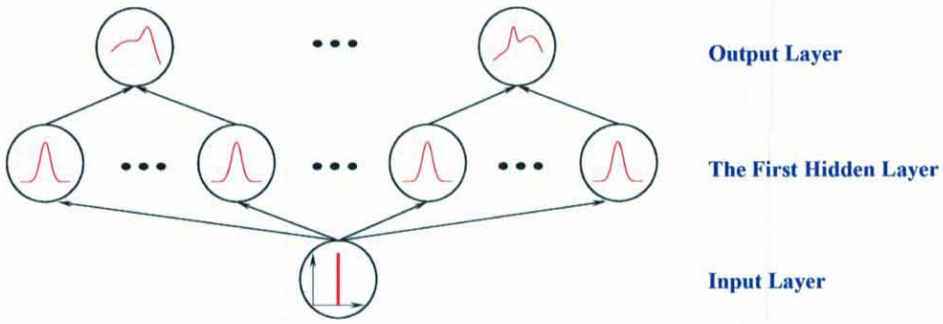


Figure 6.3: A three-layer PNN, including an input layer, a hidden layer and an output layer.

It can be seen that PNNs work using a probabilistic mixture method. Given an input signal, the information flow of a PNN proceeds as follows. It starts by checking all nodes in the first hidden layer, then checks all nodes in the upper hidden layers subsequently and finally ends in the output layer. The advantage of this probabilistic mixture method is precision since all hierarchical instances have been checked. However, the disadvantage is the high computational cost.

Therefore this thesis extends the PNNs by including a probabilistic summary method. This extension is based on the assumption that a unimodal distribution can be used to approximately represent a set of similar and mutually exclusive unimodal distributions. This assumption is inspired by Navalpakkam's model [39]. This thesis calls this type of probabilistic combination *probabilistic summary* and it can be illustratively shown in Figure 6.4. For example, a new Gaussian distribution can be estimated as a probabilistic summary of a set of mutually exclusive Gaussian distributions that have similar parameters of means and variances. This Gaussian distribution based probabilistic summary can be mathematically expressed as:

$$\begin{aligned} \overline{\mu} &= \textstyle\sum_i \mu_i \omega_i \\ \overline{\sigma}^2 &= \textstyle\sum_i (\sigma_i^2 + \mu_i^2)\omega_i - \overline{\mu}^2 \end{aligned}, \tag{6.24}$$

where μ_i and σ_i respectively denote the mean and STD of a Gaussian distribution indexed

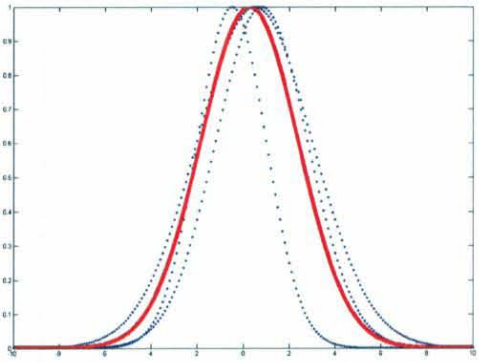


Figure 6.4: Probabilistic summary. The three blue curves respectively represent three Gaussian distributions that are similar and mutually exclusive to each other. The red curve represents the probabilistic summary (i.e., a new Gaussian distribution) of these three Gaussian distributions.

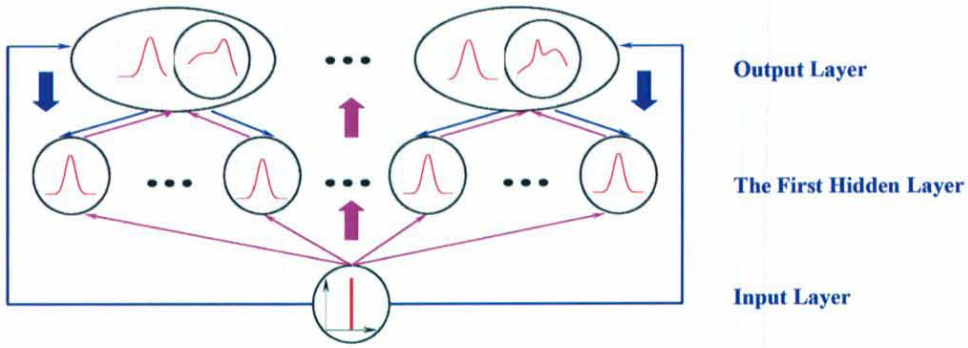


Figure 6.5: A three-layer extended PNN. It includes an input layer, a hidden layer and an output layer. The pink arrows represent the information flow using the probabilistic mixture method and the blue arrows represent the information flow using the probabilistic summary method.

by i, ω_i denotes the weight of the Gaussian distribution index by i, $\overline{\mu}$ and $\overline{\sigma}$ respectively denote the mean and STD of the new Gaussian distribution (i.e., probabilistic summary). This new Gaussian distribution is called *Gaussian summary* in this thesis.

Compared with the probabilistic mixture method, the advantage of the probabilistic summary method is that the combined distribution is unimodal so that it is computationally cheap, whereas the probabilistic summary is only an approximate estimation.

Given an input signal, the probabilistic summary can provide an opposite direction of information flow in a PNN. This opposite information flow is like a tree structure. It starts by checking all nodes in the output layer and selecting only one node as a root. Then this flow only checks the leaf nodes belonging to that root node in the lower layers subsequently. As a result, recognition in the upper coarse layers is computationally cheap. The extended PNN by combining both the probabilistic mixture method and the probabilistic summary method is shown in Figure 6.5.

The following subsections will give the local coding and the global coding by using the extended PNNs.

PNN of Local Coding

As shown in Figure 6.6, the PNN of a local coding $\mathbf{O}_{lc}$ (termed as a *local PNN*) includes four layers. The input layer receives the local post-attentive feature vector $\tilde{\mathbf{F}}_{lc}$, which includes the appearance components and salience components in terms of intensity, red-green pair, blue-yellow pair and local orientations. The first hidden layer is composed of RBFs, each of which is an instance of the learned object. Therefore, the first hidden layer of a local PNN is called an *instance layer* in this thesis. Each node of the second hidden layer is a probabilistic combination of the instance layer's RBFs that belong to the same part of that object. Therefore, the second hidden layer of a local PNN is called a *part layer*. The output layer is a probabilistic combination of all the nodes in the part layer. That is, it is the probabilistic combination of all parts belonging to the object. Therefore, the output layer of a local PNN is called an *object layer*.

Each RBF in the instance layer of a local PNN is represented by using a multi-dimensional Gaussian distribution:

$$\begin{aligned} q_i^{j,k}(\tilde{\mathbf{F}}_{lc}) &= \aleph(\tilde{\mathbf{F}}_{lc}; \boldsymbol{\mu}_i^{j,k}, \boldsymbol{\Sigma}_i^{j,k}) \\ &= \frac{1}{(2\pi)^{\frac{d}{2}} |\boldsymbol{\Sigma}_i^{j,k}|^{\frac{1}{2}}} \exp\{-\tfrac{1}{2}(\tilde{\mathbf{F}}_{lc} - \boldsymbol{\mu}_i^{j,k})^T (\boldsymbol{\Sigma}_i^{j,k})^{-1} (\tilde{\mathbf{F}}_{lc} - \boldsymbol{\mu}_i^{j,k})\} \end{aligned}, \tag{6.25}$$

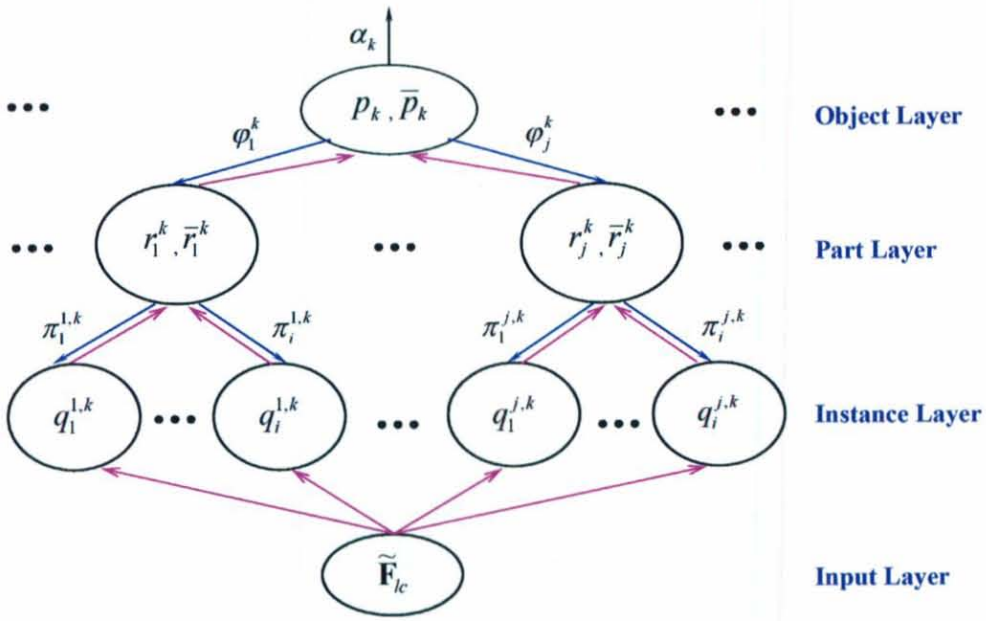


Figure 6.6: Structure of the PNN based local coding of an LTM object representation (i.e., a local PNN). i is the index of an instance belonging to a part, j is the index of a part belonging to the object, and k is the index of the LTM object representation.

where $q_i^{j,k}(\tilde{\mathbf{F}}_{lc})$ denotes the probability density of a RBF in the local PNN, $\boldsymbol{\mu}_i^{j,k}$ and $\boldsymbol{\Sigma}_i^{j,k}$ denote the mean vector and covariance matrix of a RBF in the local PNN, i is the index of an instance of a part, j is the index of a part in the local PNN of the object, k is the index of the object in LTM, and d is the dimension number of a local post-attentive feature $\tilde{\mathbf{F}}_{lc}$. Since all feature dimensions of a local coding are assumed to be independent, $\boldsymbol{\Sigma}_i^{j,k}$ is a diagonal matrix and STD values of all feature dimensions of a RBF in the local PNN can constitute an STD vector $\boldsymbol{\sigma}_i^{j,k}$.

Each node in the part layer of a local PNN represents the probability density of a part using a probabilistic combination of RBFs belonging to that part. The probabilistic mixture estimation of a node in the part layer of a local PNN can be expressed as:

$$r_j^k(\tilde{\mathbf{F}}_{lc}) = \sum_{i=1}^{N_{L1}^{lc}(j,k)} \pi_i^{j,k} q_i^{j,k}(\tilde{\mathbf{F}}_{lc}), \tag{6.26}$$

where $r_j^k(\tilde{\mathbf{F}}_{lc})$ denotes the probabilistic mixture estimation of the part j in the local PNN of the object k, $N_{L1}^{lc}(j,k)$ denotes the number of instances belonging to the part j in the local PNN of the object k, and $\pi_i^{j,k}$ denotes the occurrence rate of the instance i of the part j in the local PNN of the object k, which holds:

$$\sum_{i=1}^{N_{L1}^{lc}(j,k)} \pi_i^{j,k} = 1. \tag{6.27}$$

The probabilistic summary estimation of a node in the part layer of a local PNN can be expressed as:

$$\overline{r}_j^k(\tilde{\mathbf{F}}_{lc}) = \aleph(\tilde{\mathbf{F}}_{lc}; \overline{\boldsymbol{\mu}}_j^k, \overline{\boldsymbol{\Sigma}}_j^k), \tag{6.28}$$

where $\overline{r}_j^k(\tilde{\mathbf{F}}_{lc})$ denotes the probabilistic summary estimation of the part j in the local PNN of the object k and it is a multi-dimensional Gaussian distribution; $\overline{\boldsymbol{\mu}}_j^k$ denotes the mean vector of $\overline{r}_j^k(\tilde{\mathbf{F}}_{lc})$, and $\overline{\boldsymbol{\Sigma}}_j^k$ denotes the covariance matrix of $\overline{r}_j^k(\tilde{\mathbf{F}}_{lc})$. Since all feature dimensions of a local coding are assumed to be independent, $\overline{\boldsymbol{\Sigma}}_j^k$ is a diagonal matrix and STD values of all feature dimensions of the probabilistic summary estimation can constitute an STD vector $\overline{\boldsymbol{\sigma}}_j^k$.

The mean vector and STD vector of $\overline{r}_j^k(\tilde{\mathbf{F}}_{lc})$ can be estimated by using (6.24) and they can be expressed as:

$$\begin{aligned} \overline{\boldsymbol{\mu}}_j^k &= \textstyle\sum_{i=1}^{N_{L1}^{lc}(j,k)} \pi_i^{j,k} \boldsymbol{\mu}_i^{j,k} \\ (\overline{\sigma}_j^k)^2 &= \textstyle\sum_{i=1}^{N_{L1}^{lc}(j,k)} \left[(\sigma_i^{j,k})^2 + (\mu_i^{j,k})^2\right] \pi_i^{j,k} - (\overline{\mu}_j^k)^2 \end{aligned}, \tag{6.29}$$

where $\overline{\sigma}_j^k$ denotes an entry of the STD vector $\overline{\boldsymbol{\sigma}}_j^k$, $\sigma_i^{j,k}$ denotes an entry of the STD vector $\boldsymbol{\sigma}_i^{j,k}$, $\mu_i^{j,k}$ denotes an entry of the mean vector $\boldsymbol{\mu}_i^{j,k}$, and $\overline{\mu}_j^k$ denotes an entry of the mean vector $\overline{\boldsymbol{\mu}}_j^k$.

Each node in the object layer of a local PNN represents the probability density of the object in terms of the local features by using a probabilistic combination of the parts belonging to that object. The probabilistic mixture estimation of a node in the object

layer of a local PNN can be expressed as:

$$p_k(\tilde{\mathbf{F}}_{lc}) = \sum_{j=1}^{N_{L2}^{lc}(k)} \varphi_j^k r_j^k(\tilde{\mathbf{F}}_{lc}), \tag{6.30}$$

where $p_k(\tilde{\mathbf{F}}_{lc})$ denotes the probabilistic mixture estimation of the object k in its local PNN, $N_{L2}^{lc}(k)$ denotes the number of parts in the local PNN of the object k, and φ_j^k denotes the contribution of a part to the object k. This thesis assumes that each part contributes equally and therefore it holds constant, i.e.:

$$\varphi_j^k = 1, \forall k, \forall j. \tag{6.31}$$

Since the parts of an object are not similar to each other, a unimodal distribution cannot be used to represent the probability of an object. However, a probabilistic summary estimation of an object in its local PNN is still proposed by using the probabilistic mixture method to combine the probabilistic summary estimations of parts. Thus, the probabilistic summary estimation of a node in the object layer of a local PNN can be expressed as:

$$\overline{p}_k(\tilde{\mathbf{F}}_{lc}) = \sum_{j=1}^{N_{L2}^{lc}(k)} \varphi_j^k \overline{r}_j^k(\tilde{\mathbf{F}}_{lc}), \tag{6.32}$$

where $\overline{p}_k(\tilde{\mathbf{F}}_{lc})$ denotes the probabilistic summary estimation of the object k in its local PNN.

The maximal number of nodes in each layer is pre-defined. $N_{L1,max}^{lc}$ denotes the maximal number of instances in a local PNN, and $N_{L2,max}^{lc}$ denotes the maximal number of parts in a local PNN.

PNN of Global Coding

As shown in Fig 6.7, the PNN for a global coding $\mathbf{O}_{gb}$ (termed as *global PNN*) also includes four layers. The input layer receives the global post-attentive feature vector

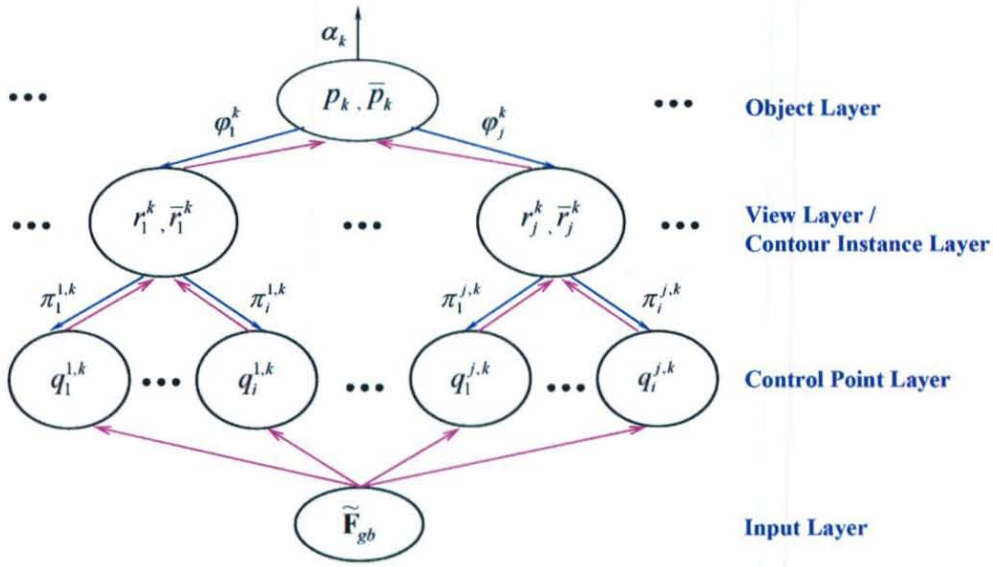


Figure 6.7: Structure of the PNN based global coding of an LTM object representation (i.e., a global PNN). i is the index of a control point belonging to a contour instance, j is the index of a contour instance belonging to the object, and k is the index of the LTM object representation.

$\tilde{\mathbf{F}}_{gb}$, which includes the appearance components and salience components of an extracted control point of the attended object. The first hidden layer is composed of RBFs, each of which is a control point along a contour instance of the LTM object. Therefore the first hidden layer of a global PNN is called a *control point layer*. Each node of the second hidden layer is a probabilistic combination of RBFs that belong to a contour instance of that object. Therefore the second hidden layer of a global PNN is called a *contour instance layer*. Since contour instances are obtained from different views, the second hidden layer of a global PNN is also called a *view layer*. The output layer is a probabilistic combination of all the nodes in the contour instance layer. That is, it is the probabilistic combination of all contour instances belonging to the object. Therefore, the output layer of a global PNN is called an *object layer*.

Each RBF in the control point layer of a global PNN is represented by using a multi-

dimensional Gaussian distribution:

$$\begin{aligned} q_i^{j,k}(\tilde{\mathbf{F}}_{gb}) &= \aleph(\tilde{\mathbf{F}}_{gb}; \boldsymbol{\mu}_i^{j,k}, \boldsymbol{\Sigma}_i^{j,k}) \\ &= \frac{1}{(2\pi)^{\frac{d}{2}} |\boldsymbol{\Sigma}_i^{j,k}|^{\frac{1}{2}}} \exp\{-\frac{1}{2}(\tilde{\mathbf{F}}_{gb} - \boldsymbol{\mu}_i^{j,k})^T (\boldsymbol{\Sigma}_i^{j,k})^{-1} (\tilde{\mathbf{F}}_{gb} - \boldsymbol{\mu}_i^{j,k})\} \end{aligned}, \tag{6.33}$$

where $q_i^{j,k}(\tilde{\mathbf{F}}_{gb})$ denotes the probability density of a RBF in the global PNN, $\boldsymbol{\mu}_i^{j,k}$ and $\boldsymbol{\Sigma}_i^{j,k}$ denote the mean vector and covariance matrix of a RBF in the global PNN, i is the index of a control point along a contour instance, j is the index of a contour instance of the global PNN of the object, k is the index of the object in LTM, and d denotes the dimension number of global post-attentive feature $\tilde{\mathbf{F}}_{gb}$. Since all feature dimensions of a global coding are assumed to be independent, $\boldsymbol{\Sigma}_i^{j,k}$ is a diagonal matrix and STD values of all feature dimensions of a RBF in the global PNN can constitute an STD vector $\boldsymbol{\sigma}_i^{j,k}$.

Each node in the contour instance layer of a global PNN represents the probability density of a contour instance by a probabilistic combination of RBFs belonging to that contour instance. The probabilistic mixture estimation of a node in the contour instance layer of a global PNN can be expressed as:

$$r_j^k(\tilde{\mathbf{F}}_{gb}) = \sum_{i=1}^{N_{L1}^{gb}(j,k)} \pi_i^{j,k} q_i^{j,k}(\tilde{\mathbf{F}}_{gb}), \tag{6.34}$$

where $r_j^k(\tilde{\mathbf{F}}_{gb})$ denotes the probabilistic mixture estimation of the contour instance j in the global PNN of the object k, $N_{L1}^{gb}(j,k)$ denotes the number of control points along the contour instance j in the global PNN of the object k, and $\pi_i^{j,k}$ denotes the occurrence rate of the control point i along the contour instance j in the global PNN of the object k, which holds:

$$\sum_{i=1}^{N_{L1}^{gb}(j,k)} \pi_i^{j,k} = 1. \tag{6.35}$$

The probabilistic summary of a node in the contour instance layer of a global PNN

can be expressed as:

$$\overline{r}_j^k(\tilde{\mathbf{F}}_{gb}) = \aleph(\tilde{\mathbf{F}}_{gb}; \overline{\boldsymbol{\mu}}_j^k, \overline{\boldsymbol{\Sigma}}_j^k), \tag{6.36}$$

where $\overline{r}_j^k(\tilde{\mathbf{F}}_{gb})$ denotes the probabilistic summary estimation of the contour instance j in the global PNN of the object k and it is a multi-dimensional Gaussian distribution; $\overline{\boldsymbol{\mu}}_j^k$ denotes the mean vector of $\overline{r}_j^k(\tilde{\mathbf{F}}_{gb})$, and $\overline{\boldsymbol{\Sigma}}_j^k$ denotes the covariance matrix of $\overline{r}_j^k(\tilde{\mathbf{F}}_{gb})$. Since all feature dimensions are assumed to be independent, $\overline{\boldsymbol{\Sigma}}_j^k$ is a diagonal matrix and STD values of all feature dimensions of $\overline{r}_j^k(\tilde{\mathbf{F}}_{gb})$ can constitute an STD vector $\overline{\boldsymbol{\sigma}}_j^k$.

The mean vector and STD vector of $\overline{r}_j^k(\tilde{\mathbf{F}}_{gb})$ can be estimated by using (6.24) and it can be expressed as:

$$\begin{aligned} \overline{\boldsymbol{\mu}}_j^k &= \textstyle\sum_{i=1}^{N_{L1}^{gb}(j,k)} \pi_i^{j,k} \boldsymbol{\mu}_i^{j,k} \\ (\overline{\sigma}_j^k)^2 &= \textstyle\sum_{i=1}^{N_{L1}^{gb}(j,k)} \left[(\sigma_i^{j,k})^2 + (\mu_i^{j,k})^2\right] \pi_i^{j,k} - (\overline{\mu}_j^k)^2 \end{aligned}, \tag{6.37}$$

where $\overline{\sigma}_j^k$ denotes an entry of the STD vector of $\overline{\boldsymbol{\sigma}}_j^k$, $\sigma_i^{j,k}$ denotes an entry of the STD vector $\boldsymbol{\sigma}_i^{j,k}$, $\mu_i^{j,k}$ denotes an entry of the mean vector $\boldsymbol{\mu}_i^{j,k}$, and $\overline{\mu}_j^k$ denotes an entry of the mean vector $\overline{\boldsymbol{\mu}}_j^k$.

One point of the probabilistic summary in the contour instance layer should be mentioned here. The appearance component of $\overline{\boldsymbol{\mu}}_j^k$ and $\overline{\boldsymbol{\sigma}}_j^k$ is not meaningful since the probabilistic summary estimation by combining the spatial positions of all control points along a contour instance cannot be used to represent the appearance distribution of that contour instance. However, the salience component of $\overline{\boldsymbol{\mu}}_j^k$ and $\overline{\boldsymbol{\sigma}}_j^k$ is meaningful since the salience distribution of a contour instance can be represented using a combination of the salience distributions of all control points along that contour instance.

Each node in the object layer of a global PNN represents the probability density of the object in terms of the global features by using a probabilistic combination of the contour instances belonging to that object. The probabilistic mixture estimation of a

node in the object layer of a global PNN can be expressed as:

$$p_k(\tilde{\mathbf{F}}_{gb}) = \sum_{j=1}^{N_{L2}^{gb}(k)} \varphi_j^k r_j^k(\tilde{\mathbf{F}}_{gb}), \tag{6.38}$$

where $p_k(\tilde{\mathbf{F}}_{gb})$ denotes the probabilistic mixture estimation of the object k in its global PNN, $N_{L2}^{gb}(k)$ denotes the number of contour instances in the global PNN of the object k, and φ_j^k denotes the contribution of a contour instance to the object k. This thesis assumes that each contour instance contributes equally and therefore it holds constant, i.e.:

$$\varphi_j^k = 1, \forall k, \forall j. \tag{6.39}$$

Since the contour instances of an object are not similar to each other, a unimodal distribution cannot be used to represent the probability of an object. However, a probabilistic summary estimation of an object in its global PNN is still proposed by using the probabilistic mixture method to combine the probabilistic summary estimations of contour instances. Thus, this probabilistic summary estimation of a node in the object layer of a global PNN can be expressed as:

$$\overline{p}_k(\tilde{\mathbf{F}}_{gb}) = \sum_{j=1}^{N_{L2}^{gb}(k)} \varphi_j^k \overline{r}_j^k(\tilde{\mathbf{F}}_{gb}), \tag{6.40}$$

where $\overline{p}_k(\tilde{\mathbf{F}}_{gb})$ denotes the probabilistic summary estimation of the object k in its global PNN.

The maximal number of nodes in each layer is also pre-defined. $N_{L1,max}^{gb}$ denotes the maximal number of control points in a global PNN, and $N_{L2,max}^{gb}$ denotes the maximal number of contour instances in a global PNN.

6.4.5 Complete Structure of LTM Object Representations

The complete structure of an LTM object representation can be expressed as follows.

It can be seen that either a local coding or a global coding consists of three levels. The local coding includes an object level, a part level and an instance level, whereas the global coding includes an object level, a contour instance level and a control point level.

The object level of the local coding of an LTM object representation indexed by k, denoted as $\mathbf{O}_{lc,k}^{L3}$, can be expressed as:

$$\mathbf{O}_{lc,k}^{L3} = \begin{pmatrix} p_k(\tilde{\mathbf{F}}_{lc}) \\ \overline{p}_k(\tilde{\mathbf{F}}_{lc}) \end{pmatrix}. \tag{6.41}$$

The part level of that local coding, denoted as $\mathbf{O}_{lc,k}^{L2}$, can be expressed as:

$$\mathbf{O}_{lc,k}^{L2} = \begin{pmatrix} r_1^k(\tilde{\mathbf{F}}_{lc}) & \cdots & r_j^k(\tilde{\mathbf{F}}_{lc}) & \cdots & r_{N_{L2,max}^{lc}}^k(\tilde{\mathbf{F}}_{lc}) \\ \overline{r}_1^k(\tilde{\mathbf{F}}_{lc}) & \cdots & \overline{r}_j^k(\tilde{\mathbf{F}}_{lc}) & \cdots & \overline{r}_{N_{L2,max}^{lc}}^k(\tilde{\mathbf{F}}_{lc}) \end{pmatrix}. \tag{6.42}$$

The instance level of that local coding, denoted as $\mathbf{O}_{lc,k}^{L1}$, can be expressed as:

$$\mathbf{O}_{lc,k}^{L1} = \begin{pmatrix} q_1^{1,k}(\tilde{\mathbf{F}}_{lc}) & \cdots & q_i^{1,k}(\tilde{\mathbf{F}}_{lc}) & \cdots & q_{N_{L1,max}^{lc}}^{1,k}(\tilde{\mathbf{F}}_{lc}) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ q_1^{j,k}(\tilde{\mathbf{F}}_{lc}) & \cdots & q_i^{j,k}(\tilde{\mathbf{F}}_{lc}) & \cdots & q_{N_{L1,max}^{lc}}^{j,k}(\tilde{\mathbf{F}}_{lc}) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ q_1^{N_{L2,max}^{lc},k}(\tilde{\mathbf{F}}_{lc}) & \cdots & q_i^{N_{L2,max}^{lc},k}(\tilde{\mathbf{F}}_{lc}) & \cdots & q_{N_{L1,max}^{lc}}^{N_{L2,max}^{lc},k}(\tilde{\mathbf{F}}_{lc}) \end{pmatrix}^T. \tag{6.43}$$

In the representations of the part level and the instance level, the remaining entries are set as unavailable when the current number of nodes is smaller than the maximum.

The object level of the global coding of an LTM object representation indexed by k, denoted as $\mathbf{O}_{gb,k}^{L3}$, can be expressed as:

$$\mathbf{O}_{gb,k}^{L3} = \begin{pmatrix} p_k(\tilde{\mathbf{F}}_{gb}) \\ \overline{p}_k(\tilde{\mathbf{F}}_{gb}) \end{pmatrix}. \tag{6.44}$$

The contour instance level of that global coding can be expressed as:

$$\mathbf{O}_{gb,k}^{L2} = \begin{pmatrix} r_1^k(\tilde{\mathbf{F}}_{gb}) & \cdots & r_j^k(\tilde{\mathbf{F}}_{gb}) & \cdots & r_{N_{L2,max}^{gb}}^k(\tilde{\mathbf{F}}_{gb}) \\ \overline{r}_1^k(\tilde{\mathbf{F}}_{gb}) & \cdots & \overline{r}_j^k(\tilde{\mathbf{F}}_{gb}) & \cdots & \overline{r}_{N_{L2,max}^{gb}}^k(\tilde{\mathbf{F}}_{gb}) \end{pmatrix}. \tag{6.45}$$

The control point level of that global coding can be expressed as:

$$\mathbf{O}_{gb,k}^{L1} = \begin{pmatrix} q_1^{1,k}(\tilde{\mathbf{F}}_{gb}) & \cdots & q_i^{1,k}(\tilde{\mathbf{F}}_{gb}) & \cdots & q_{N_{L1,max}^{gb}}^{1,k}(\tilde{\mathbf{F}}_{gb}) \\ & \cdots & \cdots & \cdots & \cdots & \cdots \\ q_1^{j,k}(\tilde{\mathbf{F}}_{gb}) & \cdots & q_i^{j,k}(\tilde{\mathbf{F}}_{gb}) & \cdots & q_{N_{L1,max}^{gb}}^{j,k}(\tilde{\mathbf{F}}_{gb}) \\ & \cdots & \cdots & \cdots & \cdots & \cdots \\ q_1^{N_{L2,max}^{gb},k}(\tilde{\mathbf{F}}_{gb}) & \cdots & q_i^{N_{L2,max}^{gb},k}(\tilde{\mathbf{F}}_{gb}) & \cdots & q_{N_{L1,max}^{gb}}^{N_{L2,max}^{gb},k}(\tilde{\mathbf{F}}_{gb}) \end{pmatrix}^T. \tag{6.46}$$

In the representations of the contour instance level and the control point level, the remaining entries are set as unavailable when the current number of nodes is smaller than the maximum.

The LTM object representation with this complete structure is also called the *high-level LTM object representation.*

6.4.6 Low-level LTM Object Representations

The complete structure of LTM object representations is used for high-level analysis only on the attended object during the post-attentive perception stage in order to produce an appropriate action. In other words, the complete structure is used to fulfill the second function of LTM object representations, i.e., guiding the action selection. However, the first function of LTM object representations is to quickly and efficiently guide top-down biasing over the whole input scene. Therefore a low-level, degraded version of the LTM object representation is required for the attentional selection stage.

Low-level Local Coding

It is evident that a rule for building a low-level object representation is to use the probabilistic summary estimation.

Furthermore, as shown in (6.5), (6.6) and (6.7), each appearance component $\tilde{\mathbf{F}}^{a}_{int}$, $\tilde{\mathbf{F}}^{a}_{rg}$ and $\tilde{\mathbf{F}}^{a}_{by}$ includes two types of statistical measures, i.e., mean and histogram. Since top-down biasing performs based on the pre-attentive features, the mean measures are used to build the appearance components of the low-level local coding.

Finally, only the part level of the high-level local coding is used to build the low-level local coding based on the fact that the proposed top-down biasing method uses a salient part to estimate the top-down biases in terms of local features.

Thus, the low-level local coding of an LTM object representation can be expressed as:

$$\mathbf{O}^{ld}_{lc,k} = \left(\begin{array}{ccccc} \overline{r}^{k}_{1}(\tilde{\mathbf{F}}^{ld}_{lc}) & \cdots & \overline{r}^{k}_{j}(\tilde{\mathbf{F}}^{ld}_{lc}) & \cdots & \overline{r}^{k}_{N^{lc}_{L2,max}}(\tilde{\mathbf{F}}^{ld}_{lc}) \end{array} \right), \tag{6.47}$$

where $\mathbf{O}^{ld}_{lc}$ denotes the low-level local coding, and $\tilde{\mathbf{F}}^{ld}_{lc}$ is identical to $\tilde{\mathbf{F}}_{lc}$, except that appearance histogram measures are removed in $\tilde{\mathbf{F}}^{ld}_{lc}$.

Low-level Global Coding

Since the proposed top-down biasing method uses control points of a salient contour instance to estimate the top-down biases in terms of contour, the appearance descriptor at the control point level in the high-level global coding is used to build the appearance descriptor of the low-level global coding.

The salience component at the contour instance level in the high-level global coding is used to build the salience descriptor of the low-level global coding based on the fact that the proposed top-down biasing method uses the salience of a contour instance.

Thus, the low-level global coding of an LTM object representation can be expressed

as:

$$\mathbf{O}_{gb,k}^{ld} = \begin{pmatrix} \mathbf{O}_{gb,k}^{ld,a} \\ \mathbf{O}_{gb,k}^{ld,s} \end{pmatrix}, \tag{6.48}$$

where $\mathbf{O}_{gb,k}^{ld}$ denotes the low-level global coding, and

$$\begin{aligned} \mathbf{O}_{gb,k}^{ld,a} &= \mathbf{O}_{gb,k}^{L1,a} \\ \mathbf{O}_{gb,k}^{ld,s} &= \begin{pmatrix} \overline{r}_1^k(\tilde{\mathbf{F}}_{gb}^s) & \cdots & \overline{r}_j^k(\tilde{\mathbf{F}}_{gb}^s) & \cdots & \overline{r}_{N_{L2,max}^{gb}}^k(\tilde{\mathbf{F}}_{gb}^s) \end{pmatrix} \end{aligned}, \tag{6.49}$$

where $\mathbf{O}_{gb,k}^{L1,a}$ denotes appearance descriptor of $\mathbf{O}_{gb,k}^{L1}$.

It is important to note that $\mathbf{O}_{lc,k}^{ld}$ and $\mathbf{O}_{gb,k}^{ld}$ are identical to the object representation used for top-down biasing, which has been presented in section 5.3.3 in Chapter 5.

6.4.7 Learning of LTM Object Representations

Once the post-attentive features of the attended object have been extracted, they are used to update the corresponding LTM object representation or to create a new one.

Dynamical Learning

In the proposed learning procedure for a local PNN, a local post-attentive feature, i.e.,$\tilde{\mathbf{F}}_{lc} = \tilde{\mathbf{F}}_{lc}(\mathbf{R}_j^{attn})$, extracted from a proto-object in the complete region being attended is regarded as a training pattern.

In the proposed learning procedure for a global PNN, a set of global post-attentive features, i.e., $\{\tilde{\mathbf{F}}_{gb}\} = \{\tilde{\mathbf{F}}_{gb}(\mathbf{r}_{cp})\}_{\forall \mathbf{r}_{cp}}$, extracted from the complete region being attended is regarded as a training pattern.

Based on the structure of the PNN based LTM object representation, the learning procedure can be basically modeled as a process to update the mean vector and covariance matrix of each RBF as well as the weights of nodes at all layers in the local and global PNNs.

If the number of nodes at each layer is known and unchanged, the expectation-maximization (EM) algorithm is optimal for learning PNNs. However, the number of nodes (i.e., the number of instances, number of parts, number of control points, number of contour instances) might be dynamically changed during the lifelong training course in this proposed cognitive perception paradigm. Thus, inspired by a constructive training method [158], this thesis proposes a dynamical learning algorithm by using both the maximum likelihood estimation (MLE) and a Bayes' classifier to update the local and global PNNs at each moment.

This proposed dynamical learning algorithm can be summarized as follows. The Bayes' classifier is used to classify the training pattern to an existing LTM pattern. The *LTM pattern* refers to a pattern of an LTM object representation at three levels. The first level is the object level for either a local PNN or a global PNN; the second level is the part level for a local PNN or the contour instance level for a global PNN; and the third level is the instance level for a local PNN or the control point level for a global PNN. In other words, given a training pattern, it is first recognized in the object level, then recognized in the part level or contour instance level, and finally recognized in the instance level or control point level. The recognition algorithms at these three levels will be presented in section 6.5.

If the training pattern can be classified to an existing LTM pattern at the instance level in a local PNN or at the control point level in a global PNN, both appearance and salience descriptors of this existing LTM pattern are updated based on MLE. Otherwise, a new LTM pattern is created. Three thresholds τ_3, τ_2 and τ_1 are introduced to determine the minimum of the correct recognition probability to an existing object, to an existing part or an existing contour instance, and to an existing instance or an existing control point respectively. Furthermore, τ^- is introduced to avoid misclassifications at the third level, i.e., the instance level for a local PNN or the control point level for a global PNN. This parameter means that the recognition probabilities of a training pattern to incorrect existing LTM patterns are less than τ^-. Thus, τ^- is used to shrink the STD of all RBFs

after each learning routine. As a result, the recognition probabilities to all incorrect RBFs are less than τ^-.

***Unsupervised and Supervised Learning*:** The above presentation has shown that the proposed dynamical learning is an unsupervised learning procedure. In addition, this proposed learning algorithm also supports supervised learning. In supervised learning, the trainer teaches the system what and/or where the attended object is. That is, a set of pairs $(\tilde{\mathbf{F}}, k)$ is given. As a result, recognition at the object level is not required in supervised learning.

Routines of the Learning Algorithm

Algorithm 1 shows the routine of the unsupervised learning algorithm for local PNNs. In this algorithm, $\overline{\overline{p}}_k(\tilde{\mathbf{F}}_{lc})$, $\overline{\overline{r}}_j^k(\tilde{\mathbf{F}}_{lc})$ and $\overline{\overline{q}}_i^{j,k}(\tilde{\mathbf{F}}_{lc})$ respectively denote the recognition probabilities of the training pattern $\tilde{\mathbf{F}}_{lc}$ at the object level, at the part level and at the instance level of a local PNN. Calculation of these recognition probabilities will be shown in section 6.5. In Algorithm 1, d denotes the dimensions of a $\tilde{\mathbf{F}}_{lc}$, $a_i^{j,k}$ denotes the occurrence number of the instance i of the part j in the local PNN of the object k, $\boldsymbol{\sigma}_{init}^{lc}$ denotes the predefined initial STD vector of a new RBF in the updated local PNN, and N_{L3} denotes the number of existing LTM objects.

It is important to note that the recognition at the object level and the recognition at the part level, i.e., steps $2 \sim 5$ in Algorithm 1, have been carried out in the perceptual completion processing module. This can be seen clearly in Figure 6.1. In order to show the entire flowchart of the learning algorithm, these two recognition procedures are also included in Algorithm 1.

Algorithm 2 shows the routine of the unsupervised learning algorithm for global PNNs. The set of global post-attentive features extracted in the complete region being attended is used as a united training pattern for recognition at the object level and at the contour instance level. Once the united training pattern is classified to an existing LTM pattern

Algorithm 1 Unsupervised Learning Routine of Local PNNs

Given a local training pattern $\tilde{\mathbf{F}}_{lc} = \tilde{\mathbf{F}}_{lc}(\mathbf{R}_j^{attn})$:
Recognize $\tilde{\mathbf{F}}_{lc}$ at the object level to obtain a recognition probability $\overline{\overline{p}}_k(\tilde{\mathbf{F}}_{lc})$;
if $\overline{\overline{p}}_k(\tilde{\mathbf{F}}_{lc}) \geq \tau_3$ **then**
    Recognize $\tilde{\mathbf{F}}_{lc}$ at the part level only in object k to obtain a recognition probability $\overline{\overline{r}}_j^k(\tilde{\mathbf{F}}_{lc})$;
    if $\overline{\overline{r}}_j^k(\tilde{\mathbf{F}}_{lc}) \geq \tau_2$ **then**
        Recognize $\tilde{\mathbf{F}}_{lc}$ at the instance level only in part j of object k to obtain a recognition probability $\overline{\overline{q}}_i^{j,k}(\tilde{\mathbf{F}}_{lc})$;
        if $\overline{\overline{q}}_i^{j,k}(\tilde{\mathbf{F}}_{lc}) \geq \tau_1$ **then**
            // Update the instance i in part j of object k
            $\forall d$: $\sigma_{temp}^d = \left[a_i^{j,k}(\sigma_{i,d}^{j,k})^2 + a_i^{j,k}(\mu_{i,d}^{j,k})^2 + (\tilde{F}_{lc,d})^2\right]/(a_i^{j,k}+1)$;
            $\boldsymbol{\mu}_i^{j,k} = (a_i^{j,k}\boldsymbol{\mu}_i^{j,k} + \tilde{\mathbf{F}}_{lc})/(a_i^{j,k}+1)$;
            $\forall d$: $\sigma_{i,d}^{j,k} = \sqrt{\sigma_{temp}^d - (\mu_{i,d}^{j,k})^2}$;
            $a_i^{j,k} = a_i^{j,k} + 1$;
        else
            // Create a new instance in part j of object k
            $N_{L1}^{lc}(j,k) = N_{L1}^{lc}(j,k) + 1; \quad i = N_{L1}^{lc}(j,k)$;
            $\boldsymbol{\mu}_i^{j,k} = \tilde{\mathbf{F}}_{lc}; \quad \boldsymbol{\sigma}_i^{j,k} = \boldsymbol{\sigma}_{init}^{lc}; \quad a_i^{j,k} = 1$;
        end if
    else
        // Create a new part of object k
        $N_{L2}^{lc}(k) = N_{L2}^{lc}(k) + 1; \quad j = N_{L2}^{lc}(k); \quad N_{L1}^{lc}(j,k) = 1; \quad i = N_{L1}^{lc}(j,k)$;
        $\boldsymbol{\mu}_i^{j,k} = \tilde{\mathbf{F}}_{lc}; \quad \boldsymbol{\sigma}_i^{j,k} = \boldsymbol{\sigma}_{init}^{lc}; \quad a_i^{j,k} = 1$;
    end if
else
    // Create a new object
    $N_{L3} = N_{L3} + 1; \quad k = N_{L3}; \quad N_{L2}^{lc}(k) = 1; \quad j = N_{L2}^{lc}(k)$;
    $N_{L1}^{lc}(j,k) = 1; \quad i = N_{L1}^{lc}(j,k)$;
    $\boldsymbol{\mu}_i^{j,k} = \tilde{\mathbf{F}}_{lc}; \quad \boldsymbol{\sigma}_i^{j,k} = \boldsymbol{\sigma}_{init}^{lc}; \quad a_i^{j,k} = 1$;
end if
// shrink STD of all RBFs
$\forall(d,i,j,k)$: $\sigma_{i,d}^{j,k} = \min\left\{\sigma_{i,d}^{j,k}, \sqrt{-|\tilde{F}_{lc,d} - \mu_{i,d}^{j,k}|^2/\ln\tau^-}\right\}$
// Normalize weights π
$\forall(i,j)$: $\pi_i^{j,k} = a_i^{j,k}/\sum_i a_i^{j,k}$.

Algorithm 2 Unsupervised Learning Routine of Global PNNs

1: Given a set of global training patterns $\{\tilde{\mathbf{F}}_{gb}\} = \{\tilde{\mathbf{F}}_{gb}(\mathbf{r}_{cp})\}_{\forall \mathbf{r}_{cp}}$:
2: Recognize $\{\tilde{\mathbf{F}}_{gb}\}$ at the object level to obtain a recognition probability $\overline{\overline{p}}_k(\{\tilde{\mathbf{F}}_{gb}\})$;
3: **if** $\overline{\overline{p}}_k(\{\tilde{\mathbf{F}}_{gb}\}) \geq (\tau_3)^{N_{ptn}}$ **then**
4: Recognize $\{\tilde{\mathbf{F}}_{gb}\}$ at the contour instance level only in object k to obtain a recognition probability $\overline{\overline{r}}_j^k(\{\tilde{\mathbf{F}}_{gb}\})$;
5: **if** $\overline{\overline{r}}_j^k(\{\tilde{\mathbf{F}}_{gb}\}) \geq (\tau_2)^{N_{ptn}}$ **then**
6: **for** $\tilde{\mathbf{F}}_{gb} \in \{\tilde{\mathbf{F}}_{gb}\}$ **do**
7: Recognize $\tilde{\mathbf{F}}_{gb}$ at the control point level only along contour instance j of object k to obtain a recognition probability $\overline{\overline{q}}_i^{j,k}(\tilde{\mathbf{F}}_{gb})$;
8: **if** $\overline{\overline{q}}_i^{j,k}(\tilde{\mathbf{F}}_{gb}) \geq \tau_1$ **then**
9: *// Update the control point i along contour instance j of object k*
10: $\forall d$: $\sigma_{temp}^d = \left[b_i^{j,k}(\sigma_{i,d}^{j,k})^2 + b_i^{j,k}(\mu_{i,d}^{j,k})^2 + (\tilde{F}_{gb,d})^2\right]/(b_i^{j,k}+1)$;
11: $\boldsymbol{\mu}_i^{j,k} = (b_i^{j,k}\boldsymbol{\mu}_i^{j,k} + \tilde{\mathbf{F}}_{gb})/(b_i^{j,k}+1)$;
12: $\forall d$: $\sigma_{i,d}^{j,k} = \sqrt{\sigma_{temp}^d - (\mu_{i,d}^{j,k})^2}$;
13: $b_i^{j,k} = b_i^{j,k} + 1$;
14: **else**
15: *// Create a new control point along contour instance j of object k*
16: $N_{L1}^{gb}(j,k) = N_{L1}^{gb}(j,k) + 1; \quad i = N_{L1}^{gb}(j,k)$;
17: $\boldsymbol{\mu}_i^{j,k} = \tilde{\mathbf{F}}_{gb}; \quad \boldsymbol{\sigma}_i^{j,k} = \boldsymbol{\sigma}_{init}^{gb}; \quad b_i^{j,k} = 1$;
18: **end if**
19: **end for**
20: **else**
21: *// Create a new contour instance in object k*
22: $N_{L2}^{gb}(k) = N_{L2}^{gb}(k) + 1; \quad j = N_{L2}^{gb}(k)$;
23: **for** $\tilde{\mathbf{F}}_{gb} \in \{\tilde{\mathbf{F}}_{gb}\}$ **do**
24: $N_{L1}^{gb}(j,k) = 1; \quad i = N_{L1}^{gb}(j,k)$;
25: $\boldsymbol{\mu}_i^{j,k} = \tilde{\mathbf{F}}_{gb}; \quad \boldsymbol{\sigma}_i^{j,k} = \boldsymbol{\sigma}_{init}^{gb}; \quad b_i^{j,k} = 1$;
26: **end for**
27: **end if**
28: **else**
29: *// Create a new object*
30: $N_{L3} = N_{L3} + 1; \quad k = N_{L3}; \quad N_{L2}^{gb}(k) = 1; \quad j = N_{L2}^{gb}(k)$;
31: Do steps $23 \sim 26$;
32: **end if**
33: *// shrink STD of all RBFs*
34: $\forall (d,i,j,k)$: $\sigma_{i,d}^{j,k} = \min\left\{\sigma_{i,d}^{j,k}, \sqrt{-|\tilde{F}_{gb,d} - \mu_{i,d}^{j,k}|^2 / \ln \tau^-}\right\}$;
35: *// Normalize weights π*
36: $\forall (i,j)$: $\pi_i^{j,k} = b_i^{j,k} / \sum_i b_i^{j,k}$.

at the contour instance layer, all training patterns in that set are subsequently used for recognition at the control point level. In Algorithm 2, $\bar{\bar{p}}_k(\{\tilde{\mathbf{F}}_{gb}\})$, $\bar{\bar{r}}_j^k(\{\tilde{\mathbf{F}}_{gb}\})$ and $\bar{\bar{q}}_i^{j,k}(\tilde{\mathbf{F}}_{gb})$ respectively denote the recognition probabilities of the united training pattern $\{\tilde{\mathbf{F}}_{gb}\}$ at the object level and at the contour instance level as well as the recognition probability of a single training pattern $\tilde{\mathbf{F}}_{gb}$ at the control point level. Calculation of these recognition probabilities will be shown in section 6.5. N_{ptn} denotes the entry number of the set $\{\tilde{\mathbf{F}}_{gb}\}$. In this algorithm, d denotes the dimensions of a $\tilde{\mathbf{F}}_{gb}$, $\boldsymbol{\sigma}_{init}^{gb}$ denotes the predefined initial STD vector of a new RBF in the updated global PNN, and $b_i^{j,k}$ denotes the occurrence number of the control point i along a contour instance j in the global PNN of the object k.

Since the STD of each RBF of local PNNs and global PNNs can only shrink and never grow in the learning algorithms, it is obvious that the proposed learning algorithms are convergent given a finite training set.

6.5 Object Recognition

Based on the structure of the PNN based object representation, the object recognition module can be modeled at three hierarchical levels. The top one is the object level. The purpose of the top level is to recognize to which LTM object an attended pattern belongs. The middle one is the part level or contour instance level. Recognition at the middle level is performed given an LTM object to which the attended pattern belongs. Thus, the purpose of the middle level is to recognize to which part in a local PNN or to which contour instance in a global PNN an attended pattern belongs. The bottom one is the instance level or control point level. Recognition at the bottom level is performed given a part or a contour instance to which the attended pattern belongs. Thus, the purpose of the bottom level is to recognize to which instance in a local PNN or to which control point in a global PNN an attended pattern belongs. The *attended pattern* is a unified term to denote a local post-attentive feature $\tilde{\mathbf{F}}_{lc}$, a set of local post-attentive features $\{\tilde{\mathbf{F}}_{lc}\}$, a

global post-attentive feature $\tilde{\mathbf{F}}_{gb}$ and a set of global post-attentive features $\{\tilde{\mathbf{F}}_{gb}\}$.

At each level, object recognition can generally be modeled as a decision unit that is based on Bayes' theorem by considering the observation likelihood and prior probability of each existing LTM pattern. The LTM pattern with maximal posterior probability would be chosen.

6.5.1 Recognition at the Object Level

Recognition at the object level is computationally expensive in that it is required to explore all LTM object representations. In order to reduce the computational cost, the proposed recognition algorithm at the object level consists of two successive steps. The first step is to explore LTM object representations by using the low-level probabilistic summary estimation of the node at the object layer of each PNN as an observation likelihood. Once the LTM object with the maximal posterior probability is selected as a candidate matched LTM object, the second step is to verify whether the attended pattern belongs to the candidate by using the high-level probabilistic mixture estimation of the node at the object layer of that candidate LTM object as an observation likelihood. The algorithms of recognition in the object level for local PNNs and global PNNs are presented respectively as follows.

Recognition Algorithm at the Object Level in Local PNNs

The attended pattern, which can be processed by the recognition algorithm at the object level in local PNNs, could be a local post-attentive feature, i.e., $\tilde{\mathbf{F}}_{lc}$, or a set of local post-attentive features, i.e., $\{\tilde{\mathbf{F}}_{lc}\}$.

Assuming that the prior probability is equal for all LTM objects, the first step for recognition at the object level in local PNNs is realized as follows. It explores the probabilistic summary estimation of each node at the object level in each local PNN in LTM and then selects an LTM object that has the maximal posterior probability by using

Bayes' theorem. This step can be mathematically expressed as:

$$k_{max} = \arg\max_{k}\{\overline{p}_k(\tilde{\mathbf{F}}_{lc})\}, \tag{6.50}$$

where $\overline{p}_k(\tilde{\mathbf{F}}_{lc})$ can be obtained using (6.32) and k_{max} is the index of the LTM object that has the maximal posterior probability, i.e., the index of the candidate matched LTM object.

In order to verify whether the attended pattern belongs to the candidate matched LTM object, the truncated part of the probabilistic mixture estimation is used. As an example, the truncated part, denoted as $\tilde{p}(\mathbf{X})$, of a Gaussian distribution (with the input vector $\mathbf{X}$, the mean vector $\boldsymbol{\mu}$ and the covariance matrix $\boldsymbol{\Sigma}$) can be given as:

$$\tilde{p}(\mathbf{X}) = \exp[(\mathbf{X}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{X}-\boldsymbol{\mu})]. \tag{6.51}$$

Since the truncated part is invariant to the change of the dimension number of the input vector, it provides a uniform measure scale for comparison with the predefined threshold (e.g., τ_3). In other words, the truncated part provides a uniform measure scale for verification. Thus, the truncated part of the Gaussian probability is used to calculate the *recognition probability* in this thesis.

The recognition probability $\overline{\overline{p}}_k(\tilde{\mathbf{F}}_{lc})$ of the attended pattern $\tilde{\mathbf{F}}_{lc}$ at the object level in local PNNs can be expressed as:

$$\overline{\overline{p}}_k(\tilde{\mathbf{F}}_{lc}) = \tilde{p}_{k_{max}}(\tilde{\mathbf{F}}_{lc}), \tag{6.52}$$

where $\tilde{p}_{k_{max}}(\tilde{\mathbf{F}}_{lc})$ denotes the truncated part of the probabilistic mixture estimation $p_{k_{max}}(\tilde{\mathbf{F}}_{lc})$ that can be obtained using (6.30).

The second step for recognition at the object level in local PNNs can be mathemati-

cally expressed as:

$$\begin{cases} \mathbf{O}_{k_{max}} \text{ is the matched LTM object} & \text{if } \overline{\overline{p}}_k(\tilde{\mathbf{F}}_{lc}) \geq \tau_3 \\ \mathbf{O}_{k_{max}} \text{ is not the matched LTM object} & \text{otherwise} \end{cases}. \tag{6.53}$$

If the attended pattern is $\{\tilde{\mathbf{F}}_{lc}\}$, each $\tilde{\mathbf{F}}_{lc}$ in the set is subsequently recognized. Assuming that all $\tilde{\mathbf{F}}_{lc}$ are independent on each other, the first step for recognizing the sets $\{\tilde{\mathbf{F}}_{lc}\}$ at the object level in local PNNs can be expressed as:

$$k_{max} = \arg\max_k \{\overline{p}_k(\{\tilde{\mathbf{F}}_{lc}\})\}, \tag{6.54}$$

where $\overline{p}_k(\{\tilde{\mathbf{F}}_{lc}\}) = \prod \overline{p}_k(\tilde{\mathbf{F}}_{lc})$.

Then the recognition probability $\overline{\overline{p}}_k(\{\tilde{\mathbf{F}}_{lc}\})$ of the attended pattern $\{\tilde{\mathbf{F}}_{lc}\}$ at the object level in local PNNs can be expressed as:

$$\overline{\overline{p}}_k(\{\tilde{\mathbf{F}}_{lc}\}) = \tilde{p}_{k_{max}}(\{\tilde{\mathbf{F}}_{lc}\}), \tag{6.55}$$

where $\tilde{p}_{k_{max}}(\{\tilde{\mathbf{F}}_{lc}\}) = \prod \tilde{p}_{k_{max}}(\tilde{\mathbf{F}}_{lc})$.

The second step for recognizing the sets $\{\tilde{\mathbf{F}}_{lc}\}$ at the object level in local PNNs can be expressed as:

$$\begin{cases} \mathbf{O}_{k_{max}} \text{ is the matched object} & \text{if } \overline{\overline{p}}_k(\{\tilde{\mathbf{F}}_{lc}\}) \geq (\tau_3)^{N_{ptn}} \\ \mathbf{O}_{k_{max}} \text{ is not the matched object} & \text{otherwise} \end{cases}, \tag{6.56}$$

where N_{ptn} denotes the entry number of the set $\{\tilde{\mathbf{F}}_{lc}\}$.

Recognition Algorithm at the Object Level in Global PNNs

The attended pattern, which can be processed by the recognition algorithm at the object level in global PNNs, is a set of global post-attentive features extracted from the complete

region being attended, i.e., $\{\tilde{\mathbf{F}}_{gb}\}$.

Assuming that the prior probability is equal for all LTM objects, the first step for recognition at the object level in global PNNs is realized as follows. It explores the probabilistic summary estimation of each node at the object level in each global PNN in LTM and then selects an LTM object that has the maximal posterior probability by using Bayes' theorem. This step can be mathematically expressed as:

$$k_{max} = \arg\max_{k} \{\overline{p}_k(\{\tilde{\mathbf{F}}_{gb}\})\}, \tag{6.57}$$

where $\overline{p}_k(\{\tilde{\mathbf{F}}_{gb}\}) = \prod \overline{p}_k(\tilde{\mathbf{F}}_{gb})$ and $\overline{p}_k(\tilde{\mathbf{F}}_{gb})$ can be obtained using (6.40).

Then the recognition probability $\overline{\overline{p}}_k(\{\tilde{\mathbf{F}}_{gb}\})$ of the attended pattern $\{\tilde{\mathbf{F}}_{gb}\}$ at the object level in global PNNs can be expressed as:

$$\overline{\overline{p}}_k(\{\tilde{\mathbf{F}}_{gb}\}) = \tilde{p}_{k_{max}}(\{\tilde{\mathbf{F}}_{gb}\}), \tag{6.58}$$

where $\tilde{p}_{k_{max}}(\{\tilde{\mathbf{F}}_{gb}\}) = \prod \tilde{p}_{k_{max}}(\tilde{\mathbf{F}}_{gb})$, and $\tilde{p}_{k_{max}}(\tilde{\mathbf{F}}_{gb})$ is the truncated part of the probabilistic mixture estimation $p_{k_{max}}(\tilde{\mathbf{F}}_{gb})$ that can be obtained using (6.38).

The second step for recognition at the object level in global PNNs can be mathematically expressed as:

$$\begin{cases} \mathbf{O}_{k_{max}} \text{ is the matched object} & \text{if } \overline{\overline{p}}_k(\{\tilde{\mathbf{F}}_{gb}\}) \geq (\tau_3)^{N_{ptn}} \\ \mathbf{O}_{k_{max}} \text{ is not the matched object} & \text{otherwise} \end{cases}, \tag{6.59}$$

where N_{ptn} denotes the entry number of the set $\{\tilde{\mathbf{F}}_{gb}\}$.

Algorithm for the Combination of Local and Global PNNs

If the attended pattern includes both local post-attentive features and global post-attentive features, the first step for recognition at the object level can be expressed

as:

$$k_{max} = \arg\max_{k} \left\{ \overline{p}_k(\{\tilde{\mathbf{F}}_{lc}\}) \times \overline{p}_k(\{\tilde{\mathbf{F}}_{gb}\}) \right\}. \tag{6.60}$$

Then the recognition probability $\overline{\overline{p}}_k(\{\tilde{\mathbf{F}}_{gb}\}, \{\tilde{\mathbf{F}}_{lc}\})$ of the attended pattern $(\{\tilde{\mathbf{F}}_{gb}\}, \{\tilde{\mathbf{F}}_{lc}\})$ at the object level can be expressed as:

$$\overline{\overline{p}}_k(\{\tilde{\mathbf{F}}_{gb}\}, \{\tilde{\mathbf{F}}_{lc}\}) = \tilde{p}_{k_{max}}(\{\tilde{\mathbf{F}}_{gb}\}) \times \tilde{p}_{k_{max}}(\{\tilde{\mathbf{F}}_{lc}\}). \tag{6.61}$$

The second step for recognition at the object level can be mathematically expressed as:

$$\begin{cases} \mathbf{O}_{k_{max}} \text{ is the matched object} & \text{if } \overline{\overline{p}}_k(\{\tilde{\mathbf{F}}_{gb}\}, \{\tilde{\mathbf{F}}_{lc}\}) \geq (\tau_3)^{N_{ptn}} \\ \mathbf{O}_{k_{max}} \text{ is not the matched object} & \text{otherwise} \end{cases}, \tag{6.62}$$

where N_{ptn} denotes the total entry number of the set $\{\tilde{\mathbf{F}}_{lc}\}$ and $\{\tilde{\mathbf{F}}_{gb}\}$.

6.5.2 Recognition at the Middle Level

In order to reduce the computational cost, the proposed recognition algorithm at the middle level also consists of two successive steps. The first step is to explore the part layer or contour instance layer in the PNN of the given LTM object by using the low-level probabilistic summary estimation as an observation likelihood. Once a part node or contour instance node with the maximal posterior probability is selected as a candidate matched LTM pattern, the second step is to verify whether the attended pattern belongs to the candidate by using the high-level probabilistic mixture estimation of the candidate node as an observation likelihood. The algorithms of recognition at the middle level for local PNNs and global PNNs are presented respectively as follows.

Recognition Algorithm at the Part Level in Local PNNs

Since the objective of recognition at the part level is to classify to which part a proto-object in the complete attended region belongs, the attended pattern at the part level is a local post-attentive feature, i.e., $\tilde{\mathbf{F}}_{lc}$.

Assuming that the prior probability is equal for all parts of the given LTM object indexed by k, the first step for recognition at the part level in local PNNs can be mathematically expressed as follows by using Bayes' theorem:

$$j_{max} = \arg\max_{j}\{\overline{r}_j^k(\tilde{\mathbf{F}}_{lc})\}, \tag{6.63}$$

where $\overline{r}_j^k(\tilde{\mathbf{F}}_{lc})$ can be obtained using (6.28) and j_{max} is the index of the part that has the maximal posterior probability, i.e., the index of the candidate matched part.

Then the recognition probability $\overline{\overline{r}}_j^k(\tilde{\mathbf{F}}_{lc})$ of the attended pattern $\tilde{\mathbf{F}}_{lc}$ at the part level in local PNNs can be expressed as:

$$\overline{\overline{r}}_j^k(\tilde{\mathbf{F}}_{lc}) = \tilde{r}_{j_{max}}^k(\tilde{\mathbf{F}}_{lc}), \tag{6.64}$$

where $\tilde{r}_{j_{max}}^k(\tilde{\mathbf{F}}_{lc})$ denotes the truncated part of the probabilistic mixture estimation $r_{j_{max}}^k(\tilde{\mathbf{F}}_{lc})$ that can be obtained using (6.26).

The second step for recognition at the part level in local PNNs can be mathematically expressed as:

$$\begin{cases} j_{max} \text{ is the matched part} & \text{if } \overline{\overline{r}}_j^k(\tilde{\mathbf{F}}_{lc}) \geq \tau_2 \\ j_{max} \text{ is not the matched part} & \text{otherwise} \end{cases}. \tag{6.65}$$

Recognition Algorithm at the Contour Instance Level in Global PNNs

The attended pattern, which can be processed by the recognition algorithm at the contour instance level in global PNNs, is also a set of global post-attentive features extracted from the complete region being attended, i.e., $\{\tilde{\mathbf{F}}_{gb}\}$.

Assuming that the prior probability is equal for all contour instances in the given LTM object indexed by k, the first step for recognition at the contour instance level in global PNNs can be mathematically expressed as follows by using Bayes' theorem:

$$j_{max} = \arg\max_{j} \left\{ \overline{r}_j^k(\{\tilde{\mathbf{F}}_{gb}\}) \right\}, \tag{6.66}$$

where $\overline{r}_j^k(\{\tilde{\mathbf{F}}_{gb}\}) = \prod \overline{r}_j^k(\tilde{\mathbf{F}}_{gb})$, $\overline{r}_j^k(\tilde{\mathbf{F}}_{gb})$ can be obtained using (6.36), and j_{max} is the index of the contour instance that has the maximal posterior probability, i.e., the index of the candidate matched contour instance.

Then the recognition probability $\overline{\overline{r}}_j^k(\{\tilde{\mathbf{F}}_{gb}\})$ of the attended pattern $\{\tilde{\mathbf{F}}_{gb}\}$ at the contour instance level in global PNNs can be expressed as:

$$\overline{\overline{r}}_j^k(\{\tilde{\mathbf{F}}_{gb}\}) = \tilde{r}_{j_{max}}^k(\{\tilde{\mathbf{F}}_{gb}\}), \tag{6.67}$$

where $\tilde{r}_{j_{max}}^k(\{\tilde{\mathbf{F}}_{gb}\}) = \prod \tilde{r}_{j_{max}}^k(\tilde{\mathbf{F}}_{gb})$, and $\tilde{r}_{j_{max}}^k(\tilde{\mathbf{F}}_{gb})$ is the truncated part of the probabilistic mixture estimation $r_{j_{max}}^k(\tilde{\mathbf{F}}_{gb})$ that can be obtained using (6.34).

The second step for recognition at the contour instance level in global PNNs can be mathematically expressed as:

$$\begin{cases} j_{max} \text{ is the matched contour instance} & \text{if } \overline{\overline{r}}_j^k(\{\tilde{\mathbf{F}}_{gb}\}) \geq (\tau_2)^{N_{ptn}} \\ j_{max} \text{ is not the matched contour instance} & \text{otherwise} \end{cases}, \tag{6.68}$$

where N_{ptn} denotes the entry number of the set $\{\tilde{\mathbf{F}}_{gb}\}$.

6.5.3 Recognition at the Bottom Level

The proposed recognition algorithm at the bottom level also consists of two successive steps. The difference from the other two levels is that only the unimodal Gaussian estimation of each RBF can be used to estimate the observation likelihood in both steps at the bottom level. That is, the first step is to explore the instance layer or control

point layer of the given part or given contour instance by using the unimodal Gaussian distributions of RBFs. Once an instance node or a control point node with the maximal posterior probability is selected as a candidate matched LTM pattern, the second step is to verify whether the attended pattern belongs to the candidate. The algorithms of recognition at the bottom level for local PNNs and global PNNs are presented respectively as follows.

Recognition Algorithm at the Instance Level in Local PNNs

The attended pattern, which can be processed by the recognition algorithm at the instance level in local PNNs, is also a local post-attentive feature, i.e., $\tilde{\mathbf{F}}_{lc}$.

Assuming that the prior probability is equal for all instances of the given part indexed by (j,k), the first step for recognition at the instance level in local PNNs can be mathematically expressed as follows by using Bayes' theorem:

$$i_{max} = \arg\max_{i}\{q_i^{j,k}(\tilde{\mathbf{F}}_{lc})\}, \tag{6.69}$$

where $q_i^{j,k}(\tilde{\mathbf{F}}_{lc})$ can be obtained using (6.25), and i_{max} denotes the index of the instance that has the maximal posterior probability, i.e., the index of the candidate matched instance.

Then the recognition probability $\overline{\overline{q}}_i^{j,k}(\tilde{\mathbf{F}}_{lc})$ of the attended pattern $\tilde{\mathbf{F}}_{lc}$ at the instance level in local PNNs can be expressed as:

$$\overline{\overline{q}}_i^{j,k}(\tilde{\mathbf{F}}_{lc}) = \tilde{q}_{i_{max}}^{j,k}(\tilde{\mathbf{F}}_{lc}), \tag{6.70}$$

where $\tilde{q}_{i_{max}}^{j,k}(\tilde{\mathbf{F}}_{lc})$ is the truncated part of the probability $q_{i_{max}}^{j,k}(\tilde{\mathbf{F}}_{lc})$.

The second step for recognition at the instance level in local PNNs can be mathemat-

ically expressed as:

$$\begin{cases} i_{max} \text{ is the matched instance} & \text{if } \overline{\overline{q}}_i^{j,k}(\tilde{\mathbf{F}}_{lc}) \geq \tau_1 \\ i_{max} \text{ is not the matched instance} & \text{otherwise} \end{cases}. \tag{6.71}$$

Recognition Algorithm at the Control Point Level in Global PNNs

The attended pattern, which can be processed by the recognition algorithm at the control point level in global PNNs, is a global post-attentive feature, i.e., $\tilde{\mathbf{F}}_{gb}$.

Assuming that the prior probability is equal for all control points in the given contour instance indexed by (j,k), the first step for recognition at the control point level in global PNNs can be mathematically expressed as follows by using Bayes' theorem:

$$i_{max} = \arg\max_i \left\{ q_i^{j,k}(\tilde{\mathbf{F}}_{gb}) \right\}, \tag{6.72}$$

where $q_i^{j,k}(\tilde{\mathbf{F}}_{gb})$ can be obtained using (6.33), and i_{max} denotes the index of the control point that has the maximal posterior probability, i.e., the index of the candidate matched control point.

Then the recognition probability $\overline{\overline{q}}_i^{j,k}(\tilde{\mathbf{F}}_{gb})$ of the attended pattern $\tilde{\mathbf{F}}_{gb}$ at the control point level in global PNNs can be expressed as:

$$\overline{\overline{q}}_i^{j,k}(\tilde{\mathbf{F}}_{gb}) = \tilde{q}_{i_{max}}^{j,k}(\tilde{\mathbf{F}}_{gb}), \tag{6.73}$$

where $\tilde{q}_{i_{max}}^{j,k}(\tilde{\mathbf{F}}_{gb})$ is the truncated part of the probability $q_{i_{max}}^{j,k}(\tilde{\mathbf{F}}_{gb})$.

The second step for recognition at the control point level in global PNNs can be mathematically expressed as:

$$\begin{cases} i_{max} \text{ is the matched control point} & \text{if } \overline{\overline{q}}_i^{j,k}(\tilde{\mathbf{F}}_{gb}) \geq \tau_1 \\ i_{max} \text{ is not the matched control point} & \text{otherwise} \end{cases}. \tag{6.74}$$

6.6 Conclusion

This chapter has presented computation in the post-attentive perception stage. This chapter asserted that the main function of the post-attentive perception stage is to interpret the attended object in detail to produce an appropriate action at the current moment, to update the corresponding LTM object representation at the current moment, and to consciously guide the top-down biasing at the next moment.

Four interactive modules of the post-attentive perception stage are modeled in this thesis: perceptual completion processing, extraction of post-attentive features, development of LTM object representations and object recognition.

Based on the IC hypothesis, the perceptual completion processing module is performed around the attended proto-object to obtain the complete region being attended. Based on the fact that the complete region contains the local instances and global features, it can provide more information used for learning the LTM object representation and producing the appropriate action.

The post-attentive feature extraction module builds a statistical WM object representation of the attended object. This WM object representation includes both high-level and low-level statistics of the attended object to facilitate the following object recognition and learning.

Development of the LTM object representations is the main module in the post-attentive perception stage. A PNN based LTM object representation is proposed in this thesis. One advantage of this proposed LTM object representation is that it can probabilistically embody various instances of that object. The other advantage is that it includes two probabilistic combination methods (i.e., probabilistic mixture and probabilistic summary) so that it can be used for both high-level post-attentive analysis and low-level top-down biasing. The result is that the learned LTM representation can be used to direct top-down biasing in the attentional selection stage, perform object recognition and learning in the post-attentive perception stage and guide action selection in the further action stage. Dynamical learning algorithms are also developed for training

the PNN based LTM object representations.

Consistent with the structure of the PNN based LTM object representations, the algorithms of object recognition have been proposed for three levels of recognition, including an object level, a part level or a contour instance level, and an instance level or a control point level. These algorithms are used for the perceptual completion processing and learning of the LTM object representations.

Chapter 7

Applications for Object Detection

7.1 Introduction

One of the important robotic applications of the proposed cognitive visual perception paradigm is object detection. That is, the proposed perception paradigm can direct the robot's attention to a salient object or to an object expected by the task, and then the attended object is sent to the post-attentive perception stage to recognize what it is or to verify whether it is the expected target. The unconscious perception path (i.e., the bottom-up competition module) can be used to detect a salient object, such as a landmark, whereas the conscious perception path (i.e., the top-down biasing module) can be used to detect the task-relevant object, i.e., the expected target. The objective of this chapter is to show the effectiveness and advantages of the unconscious aspect and conscious aspect of the proposed object-based cognitive visual perception paradigm.

Thus, this chapter includes two sections to show the robotic application for object detection of this proposed perception paradigm. Section 7.2 presents the application for detecting a salient object. Section 7.3 presents the robotic application for detecting a task-relevant object.

7.2 Detecting a Salient Object

This section presents the application for detecting a salient object. The *salient object* is defined as an object that is conspicuous to others in the scene. In other words, the salient object is an unusual or unexpected object and the current task has no prediction about its occurrence.

There are three objectives in this section. The first objective is to illustrate the capability of unconscious perception of this proposed perception paradigm. The second objective is to show the advantages of using object-based visual attention for perception by comparing it with the space-based visual attention methods. The third objective is to show the advantage of integrating the contour feature into the bottom-up competition module. The result is that an object that has a conspicuous shape compared with its neighbors can be detected.

Three experiments are shown in this section, including the detection of an object that is conspicuous in colors, in local orientations and in contour respectively.

7.2.1 Experimental Setup

Since the objective of each experiment in this application is to show the effectiveness of unconscious perception in a single feature dimension, artificial images are used in these three experiments, such that the influences from other features can be removed. The frame size of all images is 640×480 pixels. In order to show the robustness of our perception paradigm, these images are obtained using different settings, including noise, spatial transformation and changes of lighting. The noisy images are manually obtained by adding salt and pepper noise patches (noise density: $0.1 \sim 0.15$, patch size: 10×10 pixels $\sim$ 15×15 pixels) into original r, g and b color channels respectively. The experimental results are compared with the results of Itti's model (i.e., space-based bottom-up attention) [38] and Sun's model (i.e., object-based bottom-up attention) [42].

7.2.2 An Object Conspicuous in Colors

The first experiment is detecting an object that is conspicuous to its neighbors in terms of colors and all other features are approximately the same between the object and its neighbors. The results of one experiment are shown in Figure 7.1. The salient object is the red ball in this experiment. Results of the proposed perception paradigm are shown in Figure 7.1(d), which indicate that the proposed perception paradigm can detect the object that is conspicuous to its neighbors in terms of colors in different settings. Results of Itti's model and Sun's model are shown in Figure 7.1(e) and Figure 7.1(f) respectively. It can be seen that Itti's model fails to detect the salient object when noise is added to the image, as shown in column 2 in Figure 7.1(e). This indicates that the proposed object-based visual perception paradigm is more robust to noise than the space-based visual perception methods.

7.2.3 An Object Conspicuous in Local Orientations

The second experiment is detecting an object that is conspicuous to its neighbors in terms of local orientations and all other features are approximately the same between the object and its neighbors. The experimental results are shown in Figure 7.2. In this experiment, the salient object is the bar that lies in the 45° direction with respect to the horizontal direction. Detection results of the proposed perception paradigm are shown in Figure 7.2(d), which indicate that the proposed perception paradigm can detect the object that is conspicuous to its neighbors in terms of local orientations in different settings. Detection results of Itti's model and Sun's model are shown in Figure 7.2(e) and Figure 7.2(f) respectively. It can be seen that Itti's model fails to detect the salient object when noise is added to the image, as shown in column 2 in Figure 7.2(e). This indicates that the proposed object-based visual perception paradigm is more robust to noise than space-based visual perception methods.

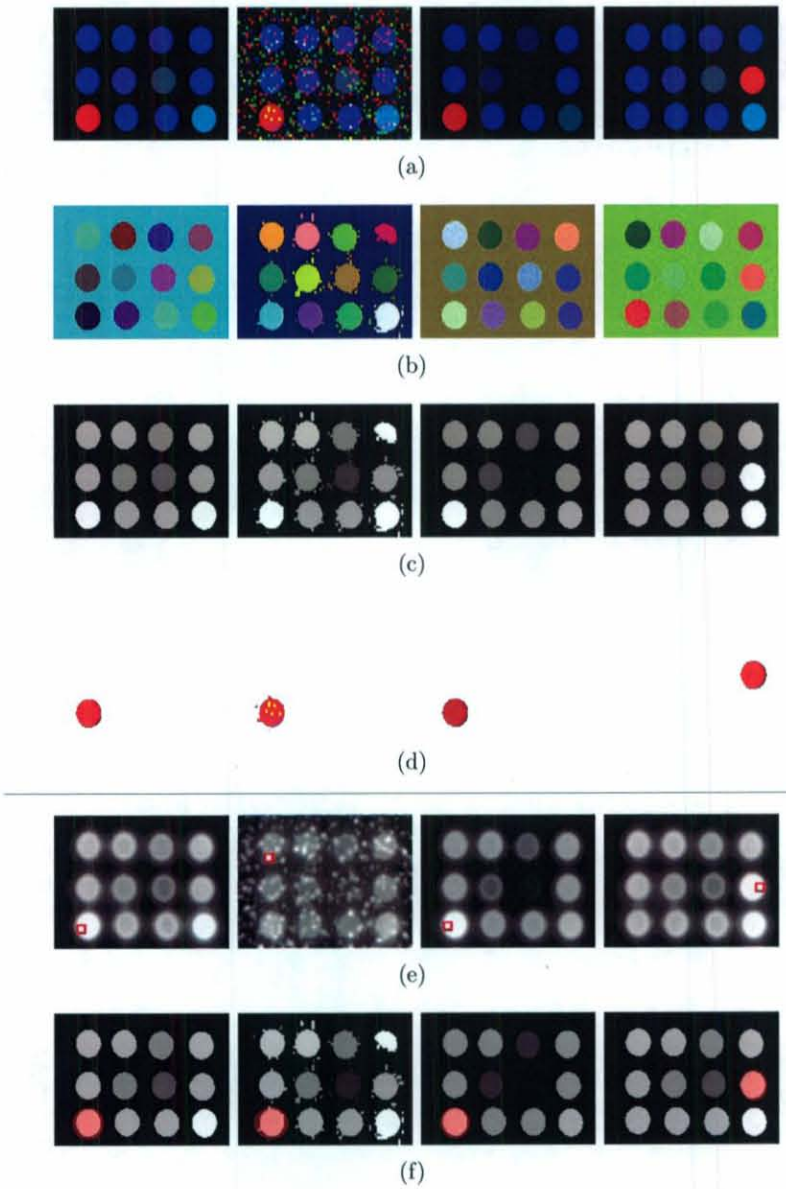


Figure 7.1: Detection of a salient object, which is conspicuous to its neighbors in terms of colors. Each column represents a type of experimental setting. Column 1 is a typical setting. Column 2 is a noise setting of column 1. Column 3 is a different lighting setting with respect to column 1. Column 4 is a spatial transformation setting with respect to column 1. Row (a): Original input images. Row (b): Pre-attentive segmentation. Each color represents one proto-object. Row (c): Proto-object based attentional activation map. Row (d): The complete region being attended. Row (e): Detection results using Itti's model. The red rectangles highlight the most salient locations. Row (f): Detection results using Sun's model. The red circles highlight the attended proto-objects.

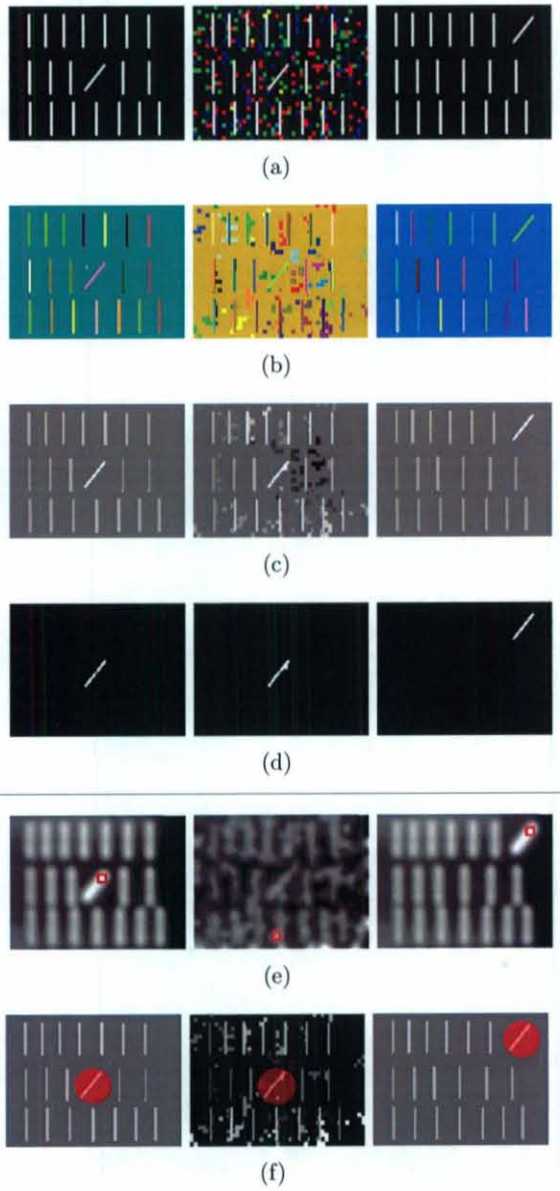


Figure 7.2: Detection of a salient object, which is conspicuous to its neighbors in terms of local orientations. Each column represents a type of experimental setting. Column 1 is a typical setting. Column 2 is a noise setting of column 1. Column 3 is a spatial transformation setting with respect to column 1. Row (a): Original input images. Row (b): Pre-attentive segmentation. Each color represents one proto-object. Row (c): Proto-object based attentional activation map. Row (d): The complete region being attended. Row (e): Detection results using Itti's model. The red rectangles highlight the most salient objects. Row (f): Detection results using Sun's model. The red circles highlight the attended proto-objects.

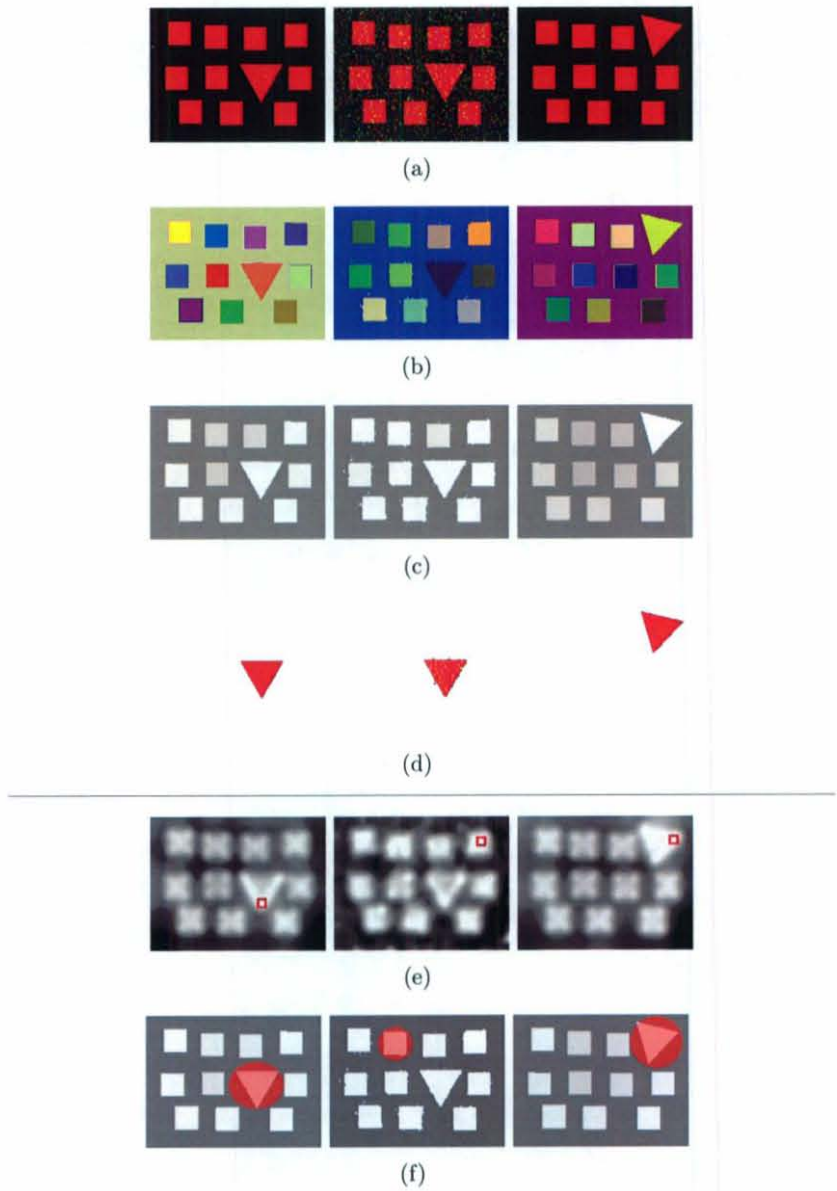


Figure 7.3: Detection of a salient object, which is conspicuous to its neighbors in terms of contour. Each column represents a type of experimental setting. Column 1 is a typical setting. Column 2 is a noise setting of column 1. Column 3 is a spatial transformation setting with respect to column 1. Row (a): Original input images. Row (b): Pre-attentive segmentation. Each color represents one proto-object. Row (c): Proto-object based attentional activation map. Row (d): The complete region being attended. Row (e): Detection results using Itti's model. The red rectangles highlight the most salient objects. Row (f): Detection results using Sun's model. The red circles highlight the attended proto-objects.

7.2.4 An Object Conspicuous in Contour

The third experiment is detecting an object that is conspicuous to its neighbors in terms of contour and all other features are approximately the same between the object and its neighbors. The experimental results are shown in Figure 7.3. In this experiment, the salient object is the triangle. Detection results of the proposed perception paradigm are shown in Figure 7.3(d), which indicate that the proposed perception paradigm can detect the object that is conspicuous to its neighbors in terms of contour in different settings. Detection results of Itti's model are shown in Figure 7.3(e) and it can be seen that Itti's model fails to detect the salient object when noise is added to the image, as shown in column 2 in Figure 7.3(e). Detection results of Sun's model are shown in Figure 7.3(f) and it can be seen that Sun's model also fails to detect the salient object when noise is added to the image, as shown in column 2 in Figure 7.3(f). This experiment indicates that the proposed object-based visual perception paradigm is capable of detecting the object conspicuous in terms of contour in different settings due to the inclusion of contour conspicuity in the proposed bottom-up competition module.

7.3 Detection of a Task-relevant Object

7.3.1 Background

It is an important ability for robots to accurately detect a task-relevant object in the cluttered environment. As presented in section 5.3.1 in Chapter 5, the *task-relevant object* is defined as an object whose occurrence is expected by the current task or defined as an object for which the current task searches. The detection of a task-relevant object is widely used in several robotic tasks, such as tracking, localization and navigation. Since most research documents use the term *object detection* to represent the detection of a task-relevant object, the term *object detection* is used in the following text of this section.

Related Work

A variety of approaches to object detection have been proposed during the past decades. The typical object detection strategy is to learn a representation of a single object or a class of objects by using a set of distinctive features and then use the learned representation to identify an instance of the single object or the class of objects in the test image. Three major components can be distinguished in the object detection systems: object representations, learning/identification algorithms and image representations.

Two categories of features, most of which are high-level, have been proposed to build object representations. The first category is global features, such as [159,160]. The second category is local features, such as edge fragments [161], rectangle features [162], Gabor filter based features [163], wavelet features [164] and interest point based features [7,165]. By using these extracted features, three categories of object representations have been proposed. The first category is point-based representations [166, 167]. In this category, both the object and the image are represented by using a set of interest points. Object detection is modeled as a matching process at these interest points. This category of methods is always used to detect a single object since the interest points are eligible to characterize a single object rather than a class of objects. The second category is global model based methods, such as [159, 168]. This category of methods attempts to match the global object representation to different regions of the test image. The third category is part-based methods, such as [169–172]. This category of methods defines a part-based object representation and attempts to find a matched instance of a part of the object in the test image. The last two categories of methods are always used to detect an instance of the class of objects since these object representations not only characterize the common properties of the objects that belong to the identical class but also are flexible enough to accommodate the within-class variability of objects.

A number of learning algorithms have been proposed, ranging from simple nearest-neighbor schemes to complex approaches, such as neural networks [173], probabilistic methods [159,169,174] and polynomial classifiers [164]. However, the problem with these

learning algorithms is that they rely on some manual steps to eliminate the background clutter.

In the area of image representations, two types of image representations have been proposed in the existing object detection methods. The first type is point-based representations [166], in which the image is regarded as a set of independent points. The second type is block-based representations [171, 172], in which the image is modeled as a set of fixed-size rectangular arrays of pixels. The problem with these two types of image representations is that completion and accuracy of the object region cannot be achieved. Recently, a region-based representation [175] has been proposed for object detection. It models an image as a set of homogeneous components using the technique of image content based segmentation. Although segmentation requires an additional computational cost, the obtained segments facilitate object detection in that each segment can eliminate distractors before identification to improve the effectiveness of identification. The segmentation procedure in this object detection method is similar to the pre-attentive segmentation module in the proposed cognitive visual perception paradigm.

Recently, a cascaded object detection method has been proposed [162, 176], which constructs a cascade of classifiers using a degenerate decision tree. The authors claim that the cascade can be seen as an object specific focus on attention mechanism. However, this detection method does not really model the visual attention mechanism.

Current Issues of Object Detection

There are three problems in this traditional strategy of object detection. The first one is that identification becomes computationally expensive in the cluttered environment since feature matching using high-level object representation is performed over the entire scene in which a large number of distractors exist.

The second problem is that it has little flexibility to use a fixed set of distinctive features. It ignores the fact that discrimination between an object and the background will be changed in different scenes and the background cannot always be specified in

advance. Thus it is possible that a set of features have to be re-designed for detecting a new class of objects or for the case that a new background shares some types of features with the object.

The third problem is perceptual completion of the object region. In most commonly studied object detection systems, e.g., [166], the training stage requires a manually segmented region of the object and the detection stage is actually a decision process to determine whether or not the input image contains an instance of the object. Although some detection systems, e.g., [171, 172], obtain a fixed-size block region of the detected object, the complete and accurate region of the object is not achieved. However, the object detection system for robots requires the ability to automatically obtain the complete and accurate object region since it can provide more useful information, e.g., shape and size of the region, for learning and producing appropriate actions.

7.3.2 The Proposed Method of Object Detection

This thesis therefore attempts to propose a new object detection method to solve the above problems by using the proposed cognitive visual perception paradigm. The proposed perception paradigm can improve the efficiency of object detection in the cluttered environment, since the attentional selection stage can serially select a candidate object using the low-level features, followed by high-level post-attentive perception only on the attended object. Therefore this thesis models object detection as a two-stage procedure. The first stage is *attentional selection*, which performs top-down biasing over the whole scene using the low-level LTM object representation. The second stage is *post-attentive recognition*, which performs only on the attended proto-object to verify whether it is the target and obtains the complete region of the target by using the high-level LTM object representation.

Since low-level features are used during the attentional selection stage in the proposed perception paradigm, the proposed object detection method is computationally efficient. Some methods have been proposed to show the effects of integrating attention

into object detection, e.g., [105, 177]. However, these methods only use the bottom-up attention. Their disadvantage is that bottom-up attention can only detect an object that is conspicuous to its neighbors. In other words, these bottom-up attention based methods cannot guide attention to the target directly when the target is not the most salient object in the scene. Thus, integrating top-down attention is an appropriate way to model object detection. The challenge of involving top-down attention is to find a compromise between effectiveness and efficiency. Traditional identification methods for object detection, e.g., [166, 171, 172], can be seen as a high-level top-down process. However, these traditional methods perform over the entire cluttered environment using high-level features and thereby they are computationally expensive. Thus, an efficient and effective top-down attention procedure using low-level distinctive features is required so that the candidate target can pop out as soon as possible during the attentional selection stage. Since the IC hypothesis is used to model top-down attention in the proposed perception paradigm, i.e., only one or a few conspicuous features of the object is deduced to effectively and efficiently guide top-down biasing (as shown in section 5.3 in Chapter 5), the proposed object detection method keeps the balance between effectiveness and efficiency for top-down attention.

As for the second issue, the proposed perception paradigm uses a set of pre-attentive feature dimensions and autonomously deduces a conspicuous feature from these dimensions for guiding top-down biasing. In other words, the learned conspicuity between the object and a variety of backgrounds is used as a metric of autonomous feature selection. Therefore the proposed object detection method is flexible enough to distinguish the object and the distractors in different scenes. That is, the proposed method is more adaptive to various scenes than other methods.

With regard to the third issue, the proposed perception paradigm provides a way to obtain a complete and accurate region of the object in the post-attentive perception stage. In fact, the completion of the object is obtained by the combination of pre-attentive segmentation and post-attentive perceptual completion processing. Pre-

attentive segmentation divides the scene into homogeneous proto-objects, one of which is then selected by attention, and finally post-attentive perceptual completion processing yields the complete and accurate object region around the attended proto-object.

Compared with other detection methods, the proposed method therefore has three advantages: 1) good balance between effectiveness and efficiency, 2) adaptivity, and 3) perceptual completion of the target region.

7.3.3 Framework of the Proposed Object Detection

According to the proposed cognitive perception paradigm, the proposed object detection method involves four modules: pre-attentive segmentation, attentional selection, post-attentive recognition and post-attentive learning of the object representation. It is illustrated in Figure 7.4.

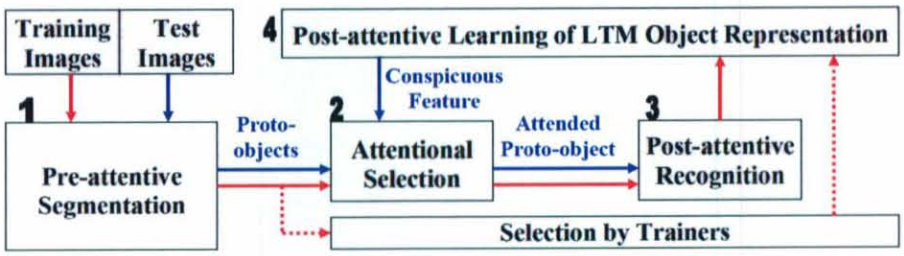


Figure 7.4: The framework of the proposed object detection method. The red lines represent learning pathways (solid lines denote the autonomous learning pathway, while dashed lines denote the interactive learning pathway) and the blue lines denote the detection pathway.

This method consists of two phases: a learning phase and a detection phase. The objective of the learning phase is to develop the LTM representation of the object to be detected. The objective of the detection phase is to detect the object in an input image by using the learned LTM representation of that object. During the first several moments, only the learning phase works. Given a training image, the complete region of the object in the image is manually selected by the trainer and it is sent to the post-attentive learning module. That is, the attentional selection module and post-attentive

recognition module do not work in these moments. Since manual selection is involved, the learning process in these moments is called the *interactive learning way*. Once the developed LTM representation of the object is capable enough to guide top-down biasing, the detection phase and learning phase start to work together. Given a test image, the object is detected by a combination of the pre-attentive segmentation module, attentional selection module and post-attentive recognition module. Then the detected object is used for learning. That is, the trainer's selection is no longer required. Thus, the learning process in these moments is called *autonomous learning way*.

The pre-attentive segmentation module divides the input image into a set of proto-objects, which are basic units of the following processing, including attentional selection, post-attentive recognition and post-attentive learning. The technical implementation of this module has been given in Chapter 4.

The attentional selection module rapidly localizes a candidate proto-object using the top-down attention mechanism. A conspicuous task-relevant feature is autonomously deduced from the low-level LTM representation of the object, as shown in (5.26) in Chapter 5. Then the task-relevant feature is used to estimate a location-based top-down bias map and the technical implementation has been given in section 5.3.6 in Chapter 5. Finally, the obtained location-based top-down bias is used to estimate the proto-object based attentional activation by subsequently using (5.61) with $w_{bu} = 0$ and $w_{td} = 1$ and using (5.62).

Once a proto-object is selected by attention, the post-attentive recognition module first obtains the complete and accurate object region around the proto-object and then recognizes the attended object in order to validate whether the attended object is the target to be detected. The detailed implementation has been given in section 6.2, section 6.3 and section 6.5 in Chapter 6. If the attended object is not the target, another procedure of attentional selection is performed by using more task-relevant features of the target.

Once the attended object is verified or the complete region of the object is selected by

the trainer, it is used to learn the corresponding LTM object representation. The learning algorithms for the local coding and the global coding have been given in Algorithm 1 and Algorithm 2 respectively in Chapter 6. In both algorithms, the index of the LTM object representation k is known.

7.3.4 Experimental Results

Experimental Setup

Three task-relevant objects are used to test the proposed object detection method: a file folder, a book and a human. For training for the file folder and the book, 20 images obtained under different viewing conditions are used respectively. For testing for the file folder and the book, 50 images obtained under different settings (including noise, transformation, lighting changes and occlusion) are respectively used. The size of each image is 640×480 pixels.

For detecting the human, three videos are obtained by a moving robot under different viewing conditions (including noise, transformation, lighting changes and occlusion). Two different office environments have been used. Video 1 and video 2 are obtained in office scene 1 with low and high lighting conditions respectively. Video 3 is obtained in office scene 2. All three videos contain a total of 650 image frames, in which 20 image frames are selected from video 1 and video 2 for training and the rest of the 630 image frames are used for testing. The size of each image is 1024×768 pixels. It is important to note that each testing image includes not only a target but also various distractors. The noise images are manually obtained by adding salt and pepper noise patches (noise density: 0.1, patch size: 5×5 pixels) into original r, g and b color channels respectively.

The results of the proposed method are compared with the results of Itti's model [38] (i.e., a space-based bottom-up attention model), Sun's model [42] (i.e., an object-based bottom-up attention model) and Navalpakkam's model [39] (i.e., a space-based top-down attention model) respectively.

Detection of An Object Having A Single Part

The first task is to detect the file folder. Due to the page limitation of the thesis, only the learning result of the low-level LTM object representation of the file folder is shown in Table 7.2. However, it is enough to show how the proposed object detection method works in this task since the low-level LTM object representation, rather than the high-level LTM object representation, is used to guide the top-down biasing that is the most important module in the proposed detection method. An example of the learning results of the high-level LTM object representation will be shown in the second task.

Table 7.2 shows that the file folder has only one part and the blue-yellow feature can be deduced as the task-relevant feature dimension since the value $\mu^s/(1+\sigma^s)$ of it is maximal. Detection results of the proposed method are shown in Figure 7.9(e). It can be seen that the file folder is successfully detected. Results of Itti's model, Sun's model and Navalpakkam's model, as shown in Figure 7.10(b), Figure 7.10(c) and Figure 7.10(d) respectively, show that these models fail to detect the target in most cases.

Detection of Objects Having Multiple Parts

The second task is to detect the book that has multiple parts. As an example, the learned high-level LTM local coding of the book in terms of the red-green pair and the learned high-level LTM global coding of the book are shown in Figure 7.7 and Figure 7.8 respectively. The learned low-level LTM representation of the book is shown in Table 7.3.

Table 7.3 has shown that the book has two parts and the blue-yellow feature in the first part can be deduced as the task-relevant feature dimension since the value $\mu^s/(1+\sigma^s)$ of this feature is maximal. Detection results of the proposed method are shown in Figure 7.11(e). It can be seen that the book is successfully detected. Results of Itti's model, Sun's model and Navalpakkam's model, as shown in Figure 7.12(b), Figure 7.12(c) and Figure 7.12(d) respectively, show that these models fail to detect the target in some cases.

The third task is to detect a human. The learning result of the low-level LTM object

representation of the human is shown in Table 7.4. It has shown that the human has two parts (including face and body) and the contour feature can be deduced as the task-relevant feature dimension since the value $\mu^s/(1+\sigma^s)$ of this feature is maximal. Detection results of the proposed method are shown in Figure 7.13(e). It can be seen that the human is successfully detected. Results of Itti's model, Sun's model and Navalpakkam's model, as shown in Figure 7.14(b), Figure 7.14(c) and Figure 7.14(d) respectively, show that these models fail to detect the target in most cases.

Performance Evaluation

Detection performance is evaluated using true positive rate (TPR) and false positive rate (FPR), which are calculated as:

$$TPR = TP/nP, \tag{7.1}$$

$$FPR = FP/nN, \tag{7.2}$$

where nP and nN are numbers of positive and negative objects respectively in the testing image set, TP and FP are numbers of true positives and false positives. The positive object is the target to be detected and the negative objects are distractors in the scene.

Detection performance of the proposed detection method and other visual attention based methods is shown in Table 7.1. Note that "Naval's" represents Navalpakkam's method in Table 7.1.

Discussion

Experimental results have shown that bottom-up attention models, e.g., [38,42], cannot detect the target successfully in most cases since they do not integrate the top-down attention mechanism. Although Navalpakkam's attention model [39] simulates the top-down attention mechanism, it is ineffective to detect the target in the environment containing

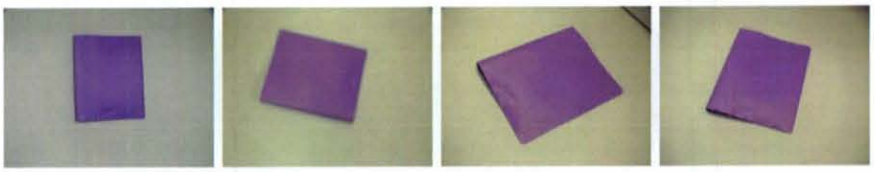
Table 7.1: Object detection performance.

Task	Method	TP	FP	nP	nN	TPR (%)	FPR (%)
1	Proposed	50	0	50	268	100.0	0.00
	Itti's	2	48	50	268	4.00	17.91
	Sun's	2	48	50	268	4.00	17.91
	Naval's	17	33	50	268	34.00	12.31
2	Proposed	47	3	50	244	94.00	1.23
	Itti's	16	34	50	244	32.00	13.93
	Sun's	27	23	50	244	54.00	9.43
	Naval's	41	9	50	244	82.00	3.69
3	Proposed	581	49	630	30949	92.22	0.16
	Itti's	5	625	630	30949	0.79	2.02
	Sun's	2	628	630	30949	0.32	2.03
	Naval's	36	594	630	30949	5.71	1.92

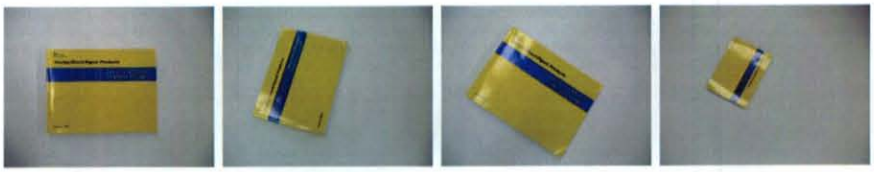
distractors which share some features with the target, as shown in Figure 7.10(d) and Figure 7.12(d), or containing a lot of clutter, as shown in Figure 7.14(d). The experimental results indicate that the proposed detection method is effective. Furthermore, the task-relevant feature(s) of the target can be selected autonomously from the learned LTM object representation. Thus the proposed detection method is adaptive to detecting any object without the requirement of pre-defining distinct types of features for different objects or scenes. Experimental results under various settings, including noise, transformation, lighting changes and occlusion, have also shown the robustness of the proposed detection method. Finally, experimental results have shown that the complete target region can be obtained in the proposed detection method.

Table 7.2: Learned low-level LTM object representation of the file folder. f denotes a pre-attentive feature dimension. n denotes the index of a part or a contour instance. The definitions of μ^a, σ^a, μ^s and σ^s can be seen in section 5.3.3 in Chapter 5.

f	n	μ^a	σ^a	μ^s	σ^s	$\mu^s/(1+\sigma^s)$
ct	1	Figure 7.6(a)		7.9	16.2	0.5
int	1	117	12.8	31.5	13.8	2.1
rg	1	3.4	9.2	63.5	28.1	2.2
by	1	25.3	7.1	191.4	5.8	**28.1**
$o_{0°}$	1	N/A	N/A	42.6	26.5	1.5
$o_{45°}$	1	N/A	N/A	38.8	18.8	2.0
$o_{90°}$	1	N/A	N/A	29.4	17.5	1.6
$o_{135°}$	1	N/A	N/A	39.1	25.0	1.5



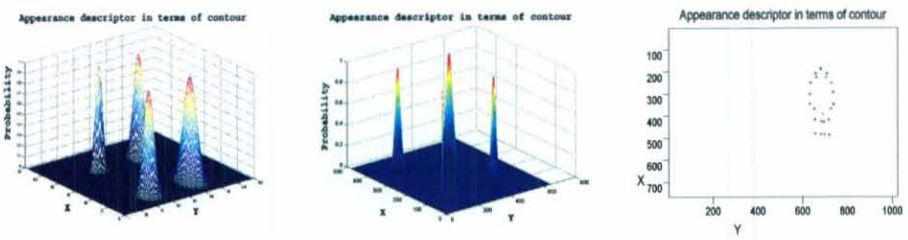
(a)



(b)

(c)

Figure 7.5: Training samples of the task-relevant objects. (a),(b),(c) Original images of some training samples of the file folder, the book and the human respectively.



(a) (b) (c)

Figure 7.6: Learned appearance descriptors in the low-level global codings. (a) The file folder. (b) The book. (c) The human (the figure is from a bird's-eye view).

Table 7.3: Learned low-level LTM object representation of the book. f denotes a pre-attentive feature dimension. n denotes the index of a part or a contour instance. The definitions of μ^a, σ^a, μ^s and σ^s can be seen in section 5.3.3 in Chapter 5.

f	n	μ^a	σ^a	μ^s	σ^s	$\mu^s/(1+\sigma^s)$
ct	1	Figure 7.6(b)		75.0	19.7	3.6
int	1	106.6	5.8	27.9	14.5	1.8
rg	1	22.1	8.7	199.6	18.2	10.4
by	1	-108.0	9.1	215.6	8.7	**22.2**
$o_{0°}$	1	N/A	N/A	41.8	9.8	3.9
$o_{45°}$	1	N/A	N/A	41.4	12.8	3.0
$o_{90°}$	1	N/A	N/A	34.7	16.3	2.0
$o_{135°}$	1	N/A	N/A	46.5	15.7	2.8
int	2	60.5	8.2	80.0	5.7	11.9
rg	2	0.4	4.3	18.3	6.4	2.5
by	2	120.8	6.7	194.7	8.1	21.4
$o_{0°}$	2	N/A	N/A	48.5	11.1	4.0
$o_{45°}$	2	N/A	N/A	53.8	9.9	4.9
$o_{90°}$	2	N/A	N/A	38.4	14.6	2.5
$o_{135°}$	2	N/A	N/A	59.4	20.3	2.8

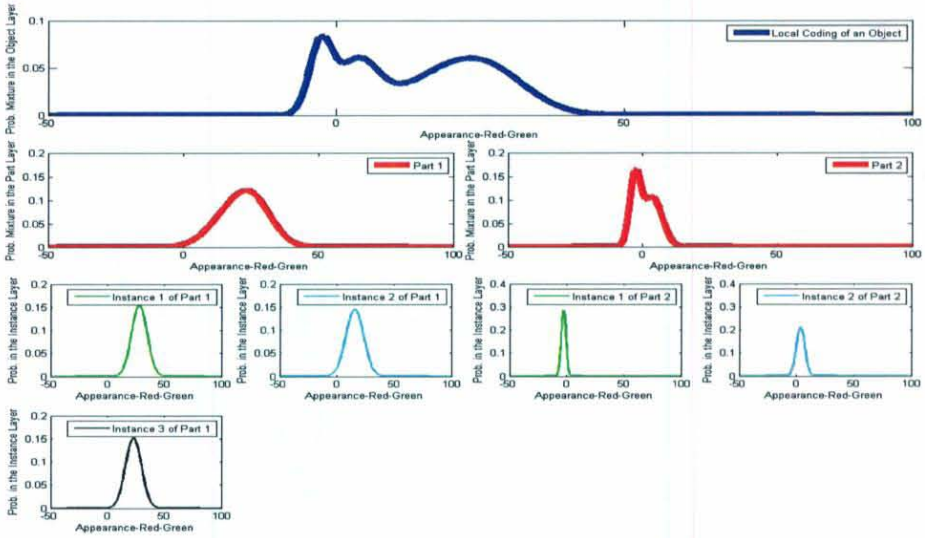
7.4 Conclusion

This chapter has presented the robotic applications of the proposed cognitive visual perception paradigm in the task of object detection. Based on this perception paradigm, object detection consists of two types. The first type is the detection of salient objects. It is based on the unconscious aspect of the proposed cognitive perception paradigm. This type of detection can be further applied for a variety of robotic tasks, such as landmark detection. Experimental results have shown the effectiveness of the bottom-up competition module and the robustness of the object-based attentional selection of the proposed cognitive perception paradigm. The second type is the detection of task-relevant objects. It is based on the conscious aspect of the proposed cognitive perception paradigm. This type of detection is implemented as a two-stage process. The first stage is attentional selection. The task-relevant feature(s) of the object to be detected are used to guide attentional selection through top-down biasing to obtain an attended proto-

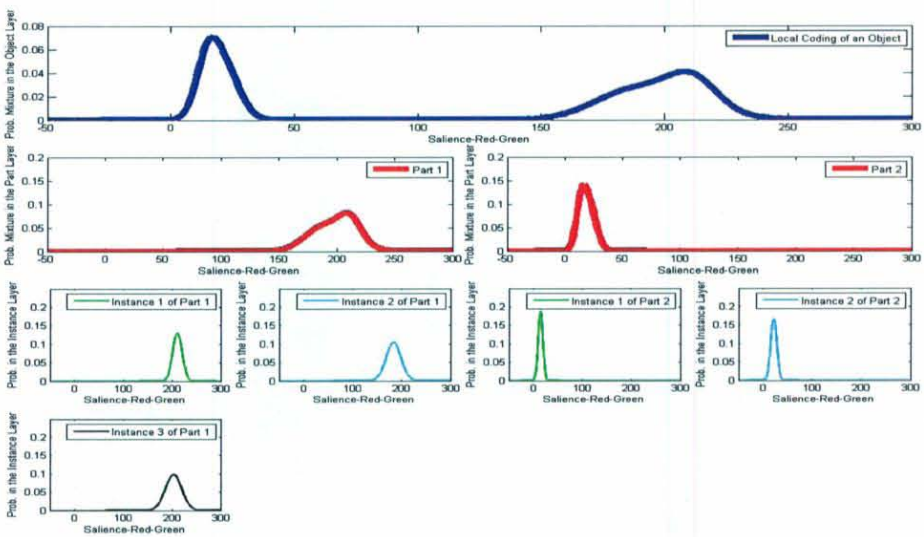
Table 7.4: Learned low-level LTM object representation of the human. f denotes a pre-attentive feature dimension. n denotes the index of a part or a contour instance. The definitions of μ^a, σ^a, μ^s and σ^s can be seen in section 5.3.3 in Chapter 5.

f	n	μ^a	σ^a	μ^s	σ^s	$\mu^s/(1+\sigma^s)$
ct	1	Figure 7.6(c)		68.3	6.9	**8.6**
int	1	28.4	21.7	18.8	13.9	1.3
rg	1	-7.0	7.1	28.6	10.8	2.4
by	1	10.9	5.4	48.4	10.9	4.1
$o_{0°}$	1	N/A	N/A	33.4	6.7	4.3
$o_{45°}$	1	N/A	N/A	39.8	11.4	3.2
$o_{90°}$	1	N/A	N/A	37.4	6.1	5.3
$o_{135°}$	1	N/A	N/A	37.5	13.5	2.6
int	2	52.0	12.5	25.6	15.6	1.5
rg	2	-2.3	17.4	49.5	18.8	2.5
by	2	-29.3	6.9	60.4	22.3	2.6
$o_{0°}$	2	N/A	N/A	12.1	6.6	1.6
$o_{45°}$	2	N/A	N/A	16.5	8.3	1.8
$o_{90°}$	2	N/A	N/A	15.0	7.9	1.7
$o_{135°}$	2	N/A	N/A	17.2	8.1	1.9

object. The second stage is post-attentive recognition. It performs only on the attended proto-object to verify whether it is the target and to obtain the complete region of the target by using the high-level LTM object representation. If not, another procedure of attentional selection is performed by using more task-relevant features. Experimental results have shown that this new two-stage object detection process is more effective, efficient, adaptive and robust than other methods.

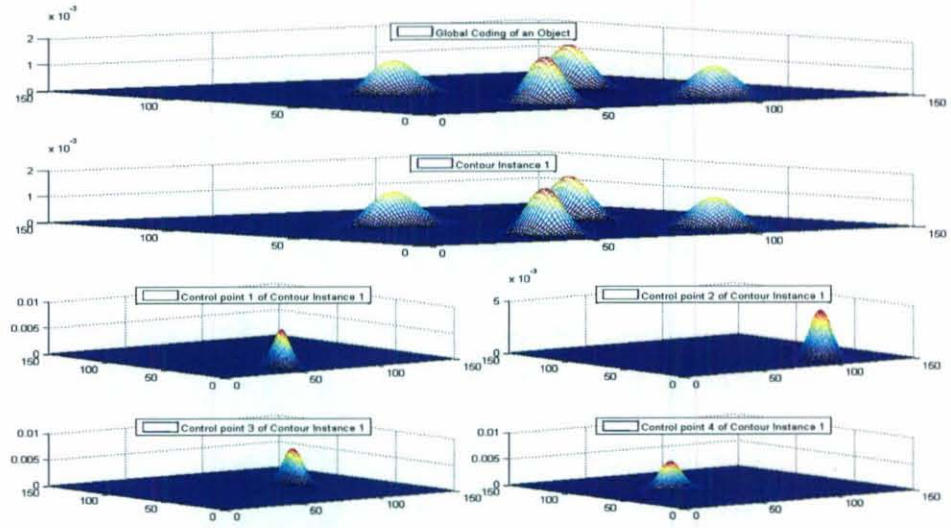


(a)

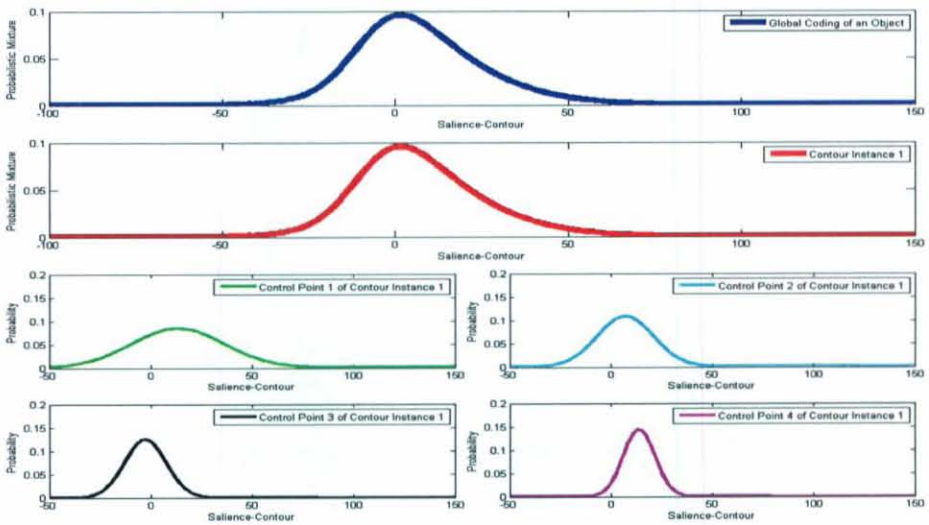


(b)

Figure 7.7: Learned high-level LTM object representation in terms of the red-green pair of the book. It can be seen that the complete structure of a PNN based local coding consists of three layers, including an object layer, a part layer and an instance layer. The curves in the instance layer show the Gaussian distribution of each instance. The curves in the part layer show the probabilistic mixture estimation of each part. The curves in the object layer show the probabilistic mixture estimation of the object. (a) Appearance descriptor. (b) Salience descriptor.



(a)



(b)

Figure 7.8: Learned high-level global coding of the LTM object representation of the book. It can be seen that the complete structure of a PNN based global coding consists of three layers, including an object layer, a contour instance layer and a control point layer. The curves in the control point layer show the Gaussian distribution of each control point. The curves in the contour instance layer show the probabilistic mixture estimation of each contour instance. The curves in the object layer show the probabilistic mixture estimation of the object. (a) Appearance descriptor. (b) Salience descriptor.

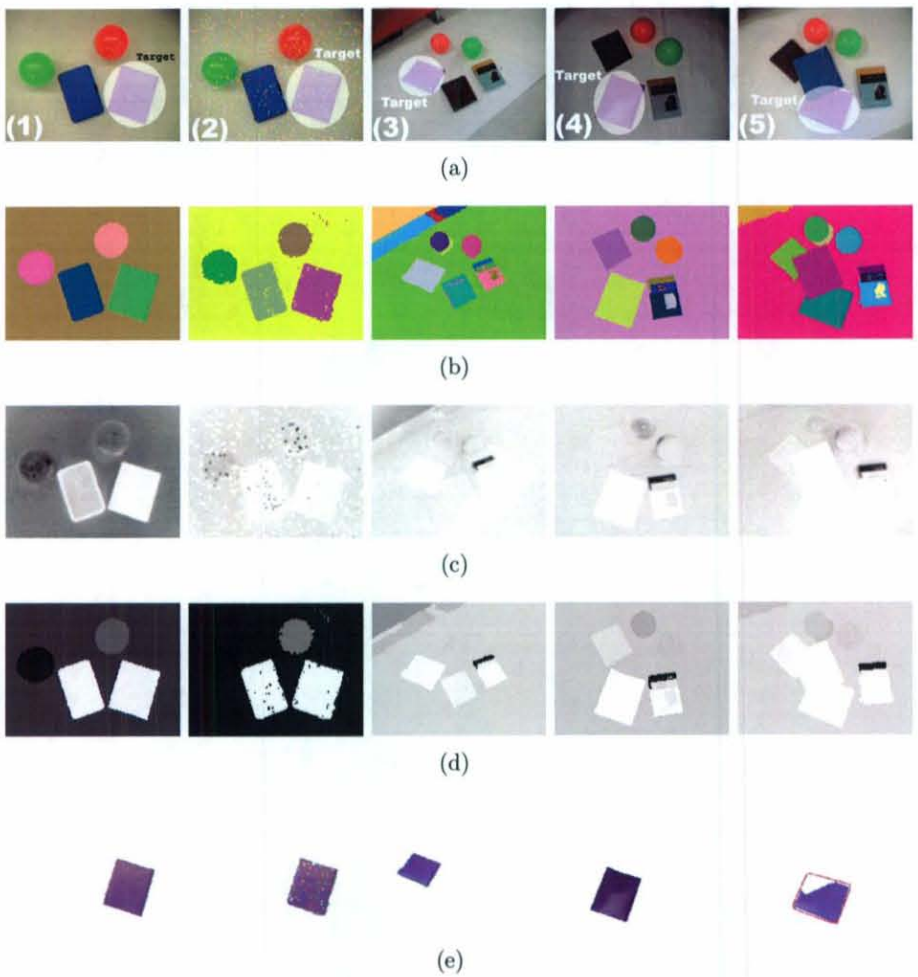


Figure 7.9: Detection of the file folder using the proposed object detection method. Each column represents a type of experimental setting. Column 1 is a typical setting. Column 2 is a noise setting of column 1. Column 3 is a spatial transformation (including translation, scaling and rotation) setting with respect to column 1. Column 4 is a different lighting setting with respect to column 1. Column 5 is an occlusion setting. Row (a): Original input images. Row (b): Pre-attentive segmentation. Each color represents one proto-object. Row (c): Location-based top-down bias map in terms of blue-yellow feature. Row (d): Proto-object based attentional activation map. Brightness represents the attentional activation value. Row (e): The complete region of the target. The red contour in the occlusion case represents the illusory contour [178], which shows the post-attentive perceptual completion effect.

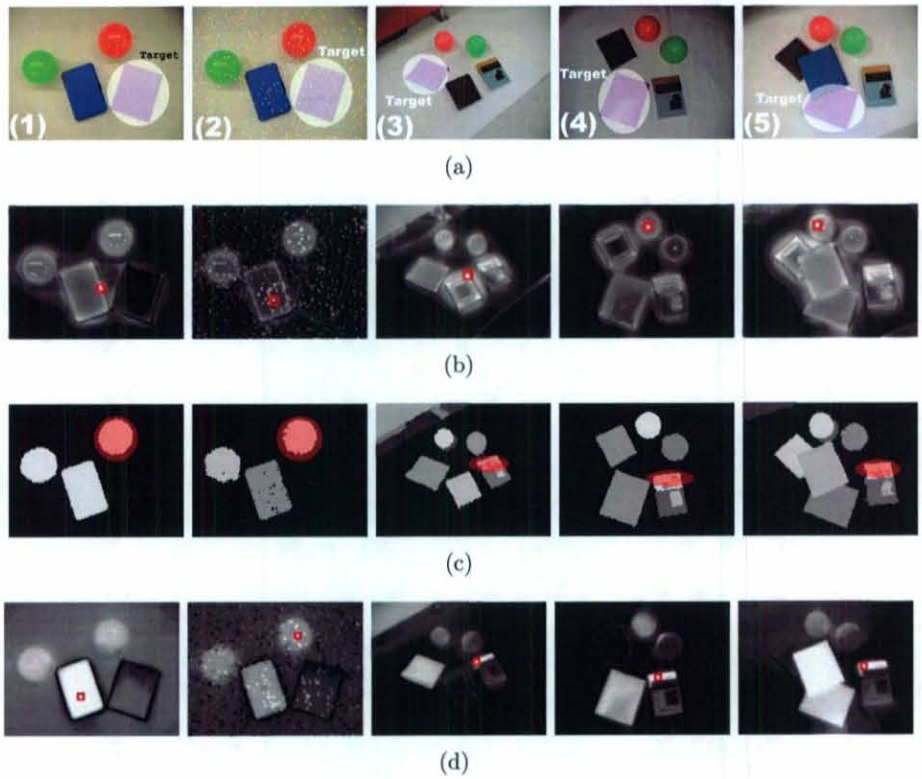


Figure 7.10: Detection of the file folder using other object detection methods. Each column represents a type of experimental setting. Column 1 is a typical setting. Column 2 is a noise setting of column 1. Column 3 is a spatial transformation (including translation, scaling and rotation) setting with respect to column 1. Column 4 is a different lighting setting with respect to column 1. Column 5 is an occlusion setting. Row (b): Detection results using Itti's model. The red rectangle highlights the most salient location. Row (c): Detection results using Sun's model. The red ellipse highlights the most salient object. Row (d): Detection results using Navalpakkam's model. The red rectangle highlights the most salient location.

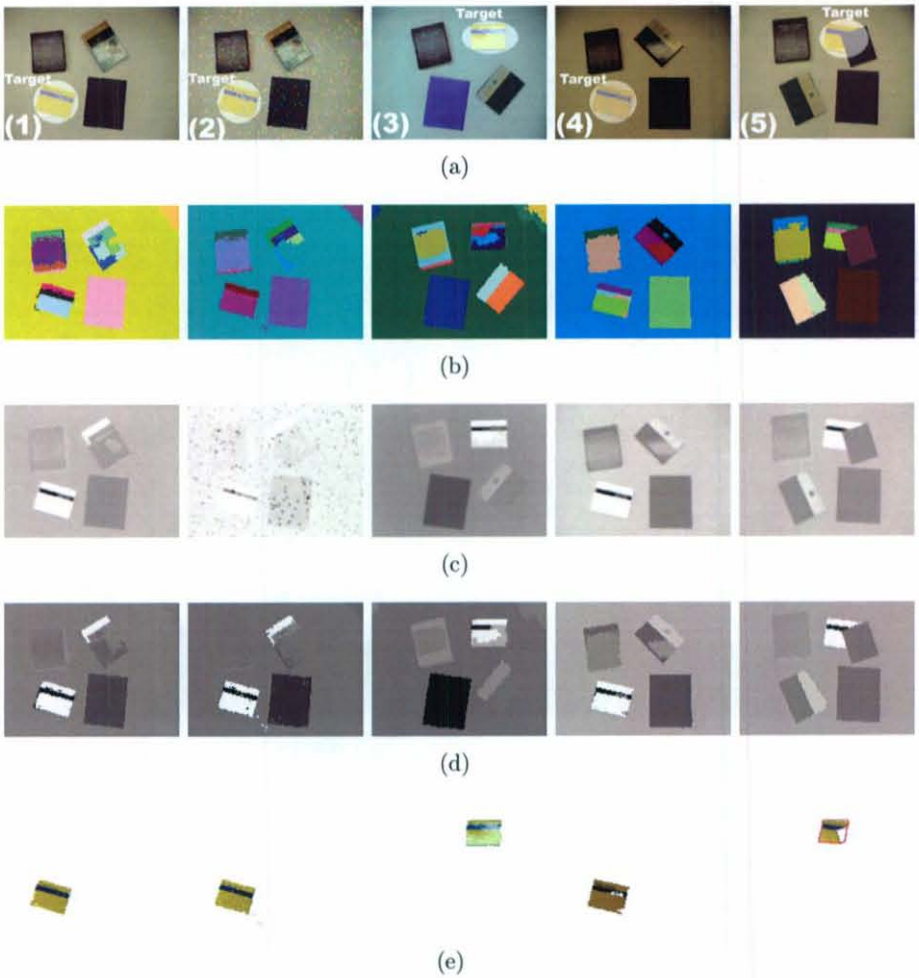


Figure 7.11: Detection of the book using the proposed object detection method. Each column represents a type of experimental setting. Column 1 is a typical setting. Column 2 is a noise setting of column 1. Column 3 is a spatial transformation (including translation and rotation) setting with respect to column 1. Column 4 is a different lighting setting with respect to column 1. Column 5 is an occlusion setting. Row (a): Original input images. Row (b): Pre-attentive segmentation. Each color represents one proto-object. Row (c): Location-based top-down bias map in terms of blue-yellow feature. Row (d): Proto-object based attentional activation map. Brightness represents the attentional activation value. Row (e): The complete region of the target. The red contour in the occlusion case represents the illusory contour [178], which shows the post-attentive perceptual completion effect.

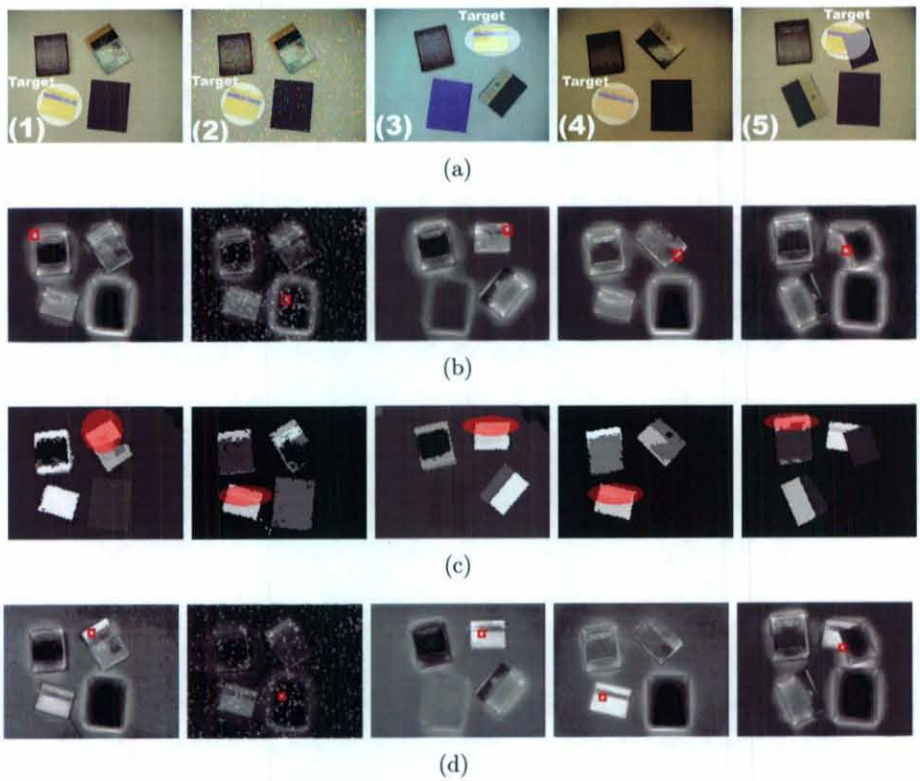


Figure 7.12: Detection of the book using other object detection methods. Each column represents a type of experimental setting. Column 1 is a typical setting. Column 2 is a noise setting of column 1. Column 3 is a spatial transformation (including translation and rotation) setting with respect to column 1. Column 4 is a different lighting setting with respect to column 1. Column 5 is an occlusion setting. Row (b): Detection results using Itti's model. The red rectangle highlights the most salient location. Row (c): Detection results using Sun's model. The red ellipse highlights the most salient object. Row (d): Detection results using Navalpakkam's model. The red rectangle highlights the most salient location.

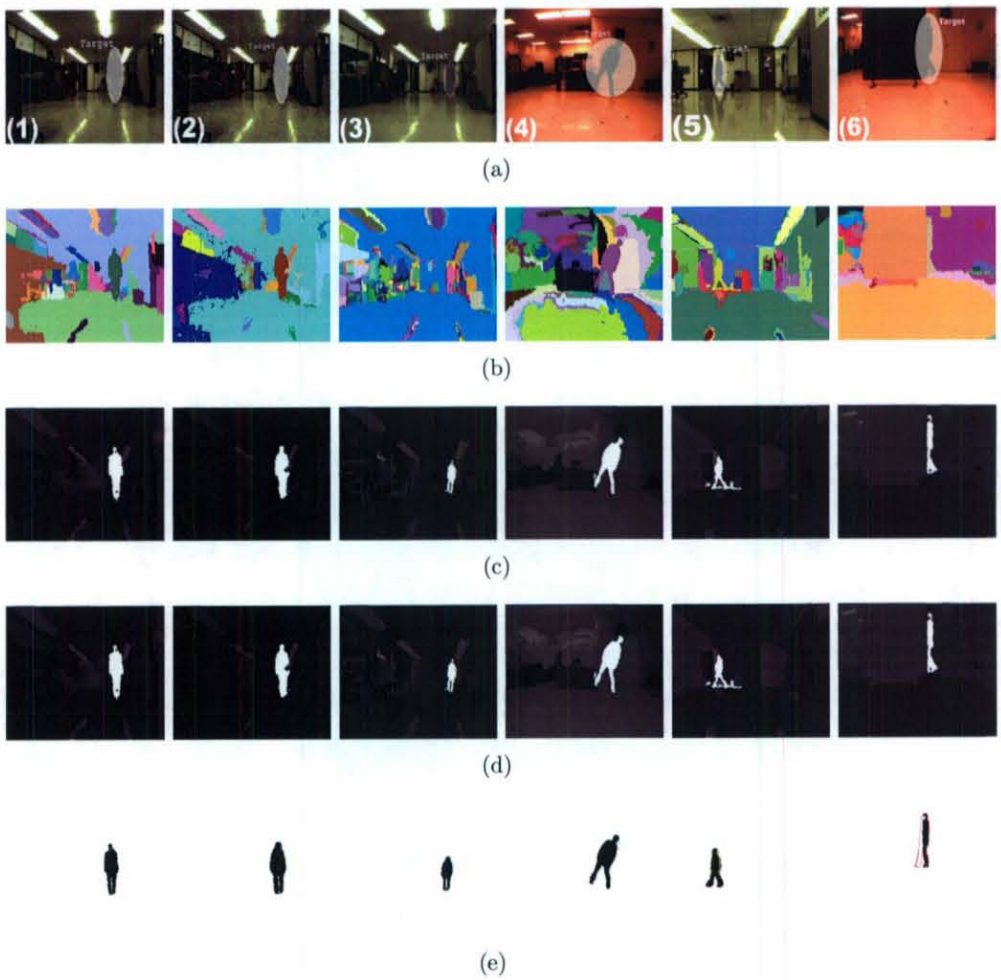


Figure 7.13: Detection of the human in the cluttered environment using the proposed object detection method. Each column represents a type of experimental setting. Column 1 is a typical setting (from video 1). Column 2 is a noise setting of column 1. Column 3 is a scaling setting with respect to column 1 (from video 1). Column 4 is a rotation setting with respect to column 1 (from video 4). Column 5 is a different lighting setting with respect to column 1 (from video 2). Column 6 is an occlusion setting (from video 4). Row (a): Original input images. Row (b): Pre-attentive segmentation. Each color represents one proto-object. Row (c): Location-based top-down bias map in terms of contour. Row (d): Proto-object based attentional activation map. Brightness represents the attentional activation value. It can be seen that these proto-object based attentional activation maps are the same with the corresponding location-based top-down bias maps. It is due to two facts. The first fact is that the units of top-down biases in terms of contour are proto-objects. The second fact is that only top-down biases contribute to the attentional activation in this task. Row (e): The complete region of the target. The red contour in the occlusion case represents the illusory contour [178], which shows the post-attentive perceptual completion effect.

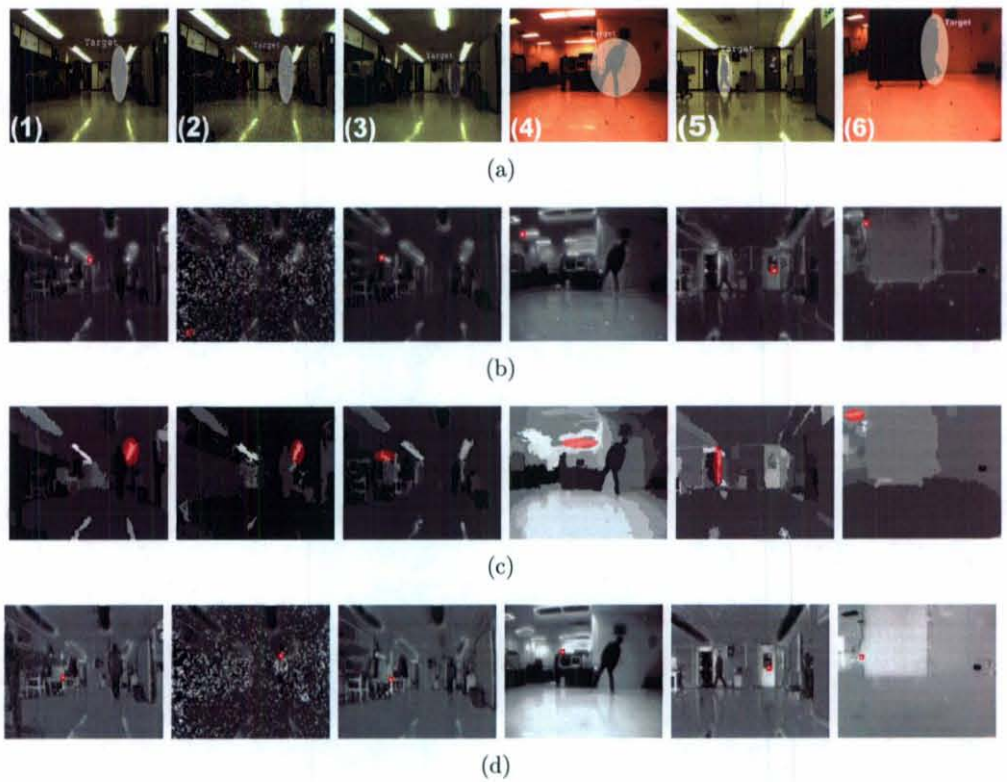


Figure 7.14: Detection of the human in the cluttered environment using other object detection methods. Each column represents a type of experimental setting. Column 1 is a typical setting (from video 1). Column 2 is a noise setting of column 1. Column 3 is a scaling setting with respect to column 1 (from video 1). Column 4 is a rotation setting with respect to column 1 (from video 4). Column 5 is a different lighting setting with respect to column 1 (from video 2). Column 6 is an occlusion setting (from video 4). Row (b): Detection results using Itti's model. The red rectangle highlights the most salient location. Row (c): Detection results using Sun's model. The red ellipse highlights the most salient object. Row (d): Detection results using Navalpakkam's model. The red rectangle highlights the most salient location.

Chapter 8

Applications for Target Tracking

8.1 Introduction

Target tracking is another important application of the proposed cognitive visual paradigm in the sense that the conscious perception path can direct the visual attention to the target to be tracked. Thus, this chapter presents the method of target tracking using the proposed cognitive visual perception paradigm.

Four sections are included in this chapter. Section 8.2 introduces some challenging issues in target tracking. Section 8.3 reviews some existing methods of target tracking. Section 8.4 presents the proposed target tracking method. Section 8.5 illustrates the experimental results using the proposed target tracking method.

8.2 Background

8.2.1 Current Issues of Target Tracking

Tracking a target of interest is an important ability for a moving robot in several types of applications, including surveillance, guiding people and driving assistance.

A typical visual tracker can be formulated as a state estimator [179]. The tracked target is defined by a state sequence that evolves over time. At each moment, the target

state is estimated by a combination of dynamical prediction and data association. Thus a tracker mainly consists of three components: target model, dynamical prediction and data association.

There are mainly three types of challenging issues in the target tracking task. The first challenging issue is caused by the cluttered and dynamically changing background. This challenge could occur in two cases: 1) Background contains a variety of clutter that shares some features with the target; 2) Discrimination between foreground and background will change dynamically during tracking and this change is especially evident in robotic applications, such as following or guiding a target on a long course. The ability to dynamically adapt the target model to the environment is one of the key points to tackle this challenge. Various features, such as contour [180–182], edge [183], optical flow [184], color [185–187], steerable pyramid [188] and Haar wavelet feature [189], have been used to build the target model. To cope with this challenging issue, two requirements for building the target model should be satisfied: robustness and discriminability. Robustness means that the target model can represent various instances of the target in different viewing conditions. Several probabilistic models, such as Gaussian distributions [185, 188] and histograms [186, 187], are proposed to improve the robustness. Subspace appearance models [190, 191] are also proposed to cope with varying pose and illumination. Discriminability means that the target model can be discriminated from the background. A method [192] is initially proposed for online selecting a discriminative feature from a set of color features by comparing likelihood variance of the target and surrounding background. This method is further extended in [193, 194]. Another method [195] is also proposed for online feature selection from a set of Haar wavelet features. However, these feature selection methods require further improvement in the following issues. Firstly, some important features are not included in the candidate feature set, such as contour. Secondly, simple geometric shapes, such as rectangles or ellipses, are used to outline the target and surrounding regions, with the result that outliers included in either region probably disturb the selection. Thirdly, the selected feature in these methods is locally

discriminative since only the background region around the target is used for comparison.

The second challenging issue is related to the recoverability of the target in the case of tracking failure. Besides background clutter, a large variation of motion and full occlusion during some sequential frames are another two major reasons causing tracking failure. Although some methods [189, 196, 197] have been proposed to accommodate abrupt motion, the occurrence of tracking failure cannot be absolutely eliminated based on the fact that the designed tracking systems cannot accommodate all possible reasons that cause failure. Thus an automatic recovery mechanism is required. This thesis proposes that two components are necessary for the recovery mechanism: validation and global search. For each frame, the estimated target state is validated. If it is inappropriate, a global search process then attempts to detect the target in the entire image. It is obvious that online selecting a target's feature that is discriminative over the entire image is required by the global search.

The third challenging issue is related to the target completion. A precise and complete target region can provide important information for a robot's following actions, such as grabbing and path planning. At present, some tracking methods only use primitive geometric shapes, such as an ellipse [186,188], to represent the target roughly. Contour based target representation [180–182,198] could achieve the completion, but it will be unsatisfactory in the cluttered environment without some segmentation processing in advance. Thus, this thesis proposes that unsupervised image segmentation that achieves homogenous subregions is helpful to achieve the target completion by subsequently applying a top-down process to these subregions. Consistent with the proposed idea, a blob-based tracking method [185] has been presented in the environment with little clutter. However, integrating an effective and efficient unsupervised segmentation into tracking for the highly cluttered environment is still an open issue.

8.2.2 Proposed Method for Target Tracking

Therefore this thesis attempts to propose a biologically-inspired visual target tracking method using the proposed cognitive visual perception paradigm in order to solve the above issues in a unified framework. The target tracking process is modeled as a three-stage process, consisting of pre-attentive segmentation, attentional selection and post-attentive recognition. The pre-attentive segmentation stage automatically divides the scene into homogeneous proto-objects. The attentional selection stage performs top-down biasing based on the task-relevant feature of the target to attentionally select one proto-object. If the attended proto-object is confirmed to be the tracked target in the post-attentive recognition stage, a complete target region is then achieved. Otherwise, it means an occurrence of tracking failure and another attentional selection process (i.e., recovery procedure) is carried out over the entire image.

By using the proposed LTM object representation and the dynamical learning algorithms, as shown in section 6.4 in Chapter 6, the target model can be built and learned online. The PPN based target model can improve the robustness in that it can embody a variety of instances of the target. The inclusion of salience descriptors in the target model can also improve the discriminability in that the salience descriptors can represent the global discriminability between the target and the background over the entire image. Furthermore, the target model is built using the pre-attentive features, including intensity, red-green, blue-yellow, local orientations and contour. Thus, this target model not only reduces computational cost due to the low-level property of these pre-attentive features but also covers a broad feature space for tracking.

The integration of the post-attentive recognition stage based on the high-level LTM representation of the target can provide a high-level and precise identification of the attended object to activate the automatic recovery mechanism for tracking failure. Meanwhile, the task-relevant feature(s) deduced from the salience descriptors of LTM target representation are globally discriminative such that they are eligible for global search during the recovery procedure.

By a combination of pre-attentive segmentation and post-attentive perceptual completion processing, the complete region of the target can be obtained. The complete target region can furthermore outline the target and background precisely to improve the selection of the task-relevant feature for the next moment.

Therefore this new tracking method has the following advantages:

1. Adaptivity: The task-relevant feature(s), which can globally discriminate the target and the background, can be autonomously deduced online from the learned salience descriptors such that they can cope with cluttered and dynamically changing environments. Furthermore, a broad feature space can be used as the candidate of the task-relevant feature(s).

2. Robustness: It has the ability to automatically recover tracking failure caused by any reason.

3. Target completion: By a combination of pre-attentive segmentation and post-attentive completion processing, the precise and complete target region is achieved.

8.3 Related Work

Target tracking from stationary cameras has been effectively achieved by using frame differencing or adaptive background subtraction techniques [179]. However, target tracking from a moving camera is a great challenge due to background motion. Egomotion estimation [199] is a popular approach to dealing with the background motion. It calculates the background's motion vector, which is then used for compensation. Most of egomotion methods are based on registering the background motion using a linear spatial transformation [200], e.g. affine transformation. These methods assume that the apparent motion of background is dominant in the image sequence. Feature matching between two consecutive images is applied to estimate parameters of the spatial transformation. Although egomotion estimation performs well in the area of computer vision, there is

one problem in robot applications: the background motion is sometimes geometrically nonlinear if images are taken from moving robots. For instance, the background motion is a radial expansion outward the center field of view when the robot moves forward.

Appearance-based methods are good candidates for coping with the challenge of tracking from a moving camera. There are mainly two categories of appearance-based methods for visual tracking. The first category is image registration based methods, such as kernel-based tracking [186], template-based tracking [183], contour-based tracking [182] and motion-based tracking [180, 188, 190, 191]. These methods are based on an image constancy assumption. The target region has no or small changes in terms of some appearance features (e.g., illumination and texture) between the present frame and previous frame. Thus, these methods attempt to optimize a correlation-like criterion, which measures the similarity between the previous state and observations in the present scene.

The second category is recursive Bayes' filter based methods, which model probabilistic representations of states, dynamics and observations and estimate the optimal state at each time step using Bayes' theorem and probabilistic estimation techniques. Thus these methods adapt to uncertainties. The Kalman filter [179] is always used to implement the recursive Bayes' filter in the case that both dynamics and observation functions are linear. The extended Kalman filter (EKF) [179] and the unscented Kalman filter (UKF) [201] are further proposed for the case that dynamics or observation functions are nonlinear. The particle filter [202] is a general approach to implement the recursive Bayes' filter by representing the density using a set of weighted samples. For tracking in the environment containing multiple targets, a probabilistic data association filter (PDAF) [179] and a joint PDAF (JPDAF) [203] are proposed. Applications of these algorithms in a variety of scenarios have been presented: [204] uses the Kalman filter for tracking vehicles; [205] and [206] use the EKF for object tracking and 3-D pose estimation; and the condensation algorithm [181] as well as other tracking algorithms [189, 207–209] are presented based on particle filters.

Some methods [210,211] are also proposed to model tracking as a foreground-background

classification procedure.

Visual attention applications for target tracking have been proposed recently. A few tracking approaches based on bottom-up attention mechanism have been reported recently [212], but these methods suffer if the background is more salient than the target. In the case of biological visual attention, humans keep the target model in their memory while tracking the object. Hence top-down biasing is necessary when tracking in a highly cluttered environment. As we know, a top-down attention based tracking approach is firstly presented by [213]. In that method, a target model is learned at the beginning of tracking in terms of low-level features according to the target's uniqueness of each feature, and tracking is modeled as a top-down visual search procedure based on the learned target model. One difference between the proposed method and [213] is that the proposed method is object based by using Duncan's IC hypothesis. The IC hypothesis provides two advantages: 1) A task-relevant feature of the target can be explicitly discriminated from the background online so as to effectively and efficiently guide top-down attentional selection, and 2) the complete target region can be achieved after attentional selection. The other difference is that the task-relevant feature can be selected from a broader feature space, including contour, intensity, colors and orientations, in the proposed method.

8.4 Framework of Proposed Tracking Method

The proposed tracking method consists of four modules as shown in Figure 8.1: Online learning of the target model, pre-attentive segmentation, attentional selection and post-attentive recognition.

The pre-attentive segmentation module extracts a set of pre-attentive features at multiple scales and then divides the scene into homogenous proto-objects in an unsupervised manner. The technical implementation of this module has been given in Chapter 4.

The target model is learned simultaneously with the tracking process, as shown using

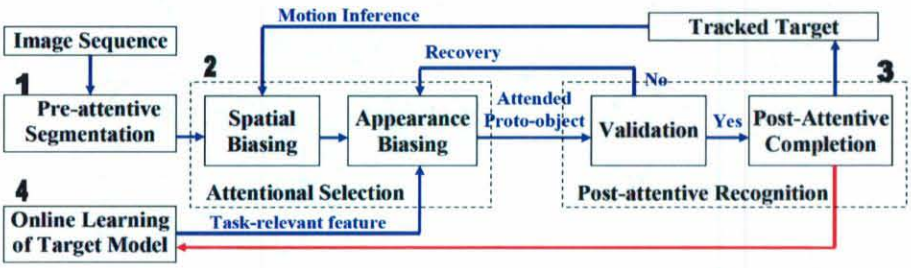


Figure 8.1: The framework of the proposed target tracking method.

the red lines in Figure 8.1. In the first tracking frame, the target model is initialized by using only one type of supervision information: the trainer specifies which proto-objects belong to the target. In the following tracking frames, the tracked instance of the target is used to update the target model so that it can accommodate changes in the environment. The learning algorithms for the local coding and the global coding have been given in Algorithm 1 and Algorithm 2 respectively in Chapter 6. In both algorithms, the index of the LTM object representation k is known. A minor change is made in both learning algorithms for the tracking task. It is that some inactive RBFs of the global PNN and the local PNN are discarded in order to keep track of the most recent target's state. A pre-defined threshold is used to determine whether a RBF is active or inactive.

Following pre-attentive segmentation, the attentional selection module, including spatial biasing (i.e., dynamical prediction) and appearance biasing (i.e., data association), is carried out. Spatial biasing estimates a spatial bias map based on the target region at previous moments and dynamical prediction techniques. Due to the variation of target's motion, target dynamics is difficult to estimate. Thus, this thesis only predicts a large region centered at the target position at the last moment as the predicted region. Appearance top-down biasing is then performed in that region. Using the task-relevant feature of that target, appearance biasing then evaluates a proto-object based attentional activation map, which represents the likelihood of each proto-object to be the tracked target in the predicted region. The proto-object with the maximal attentional activation is selected as the attended proto-object. The detailed implementation of the appearance

biasing has been given in Chapter 5.

Once the attended proto-object is obtained, it is sent to the post-attentive recognition module, including validation and post-attentive completion processing. If the attended object is confirmed to be the target, a precise and complete target region around the attended proto-object is obtained. Otherwise, it means an occurrence of tracking failure and the recovery mechanism is triggered by carrying out the appearance biasing procedure again over the entire image to globally search for the target. The detailed implementation of the post-attentive recognition and perceptual completion processing has been given in section 6.2, section 6.3 and section 6.5 in Chapter 6.

8.5 Experiments

This proposed tracking method is tested in four tasks in different scenes to show its advantages. All tracking results are shown in the attached videos.

Meanwhile, performance of the proposed method is also compared with CamShift (Continuously adaptive mean shift) algorithm [214]. Camshift algorithm is one of the appearance-based target tracking approaches. It is an adaptation of the mean shift algorithm [215], which is a non-parametric technique to find the distribution mode of the target by climbing the gradient of the probabilistic distribution.

8.5.1 Experimental Setup

Four videos are obtained by a moving robot in four different scenes under different settings, including variations of lighting and viewing conditions and occlusion. The frame size of video 1 and video 2 is 1024×768 pixels and the frame size of video 3 and video 4 is 720×576 pixels. In order to test the robustness of the proposed tracking method in cases of large variation of motion, lower frame rates are accepted in these experiments. The frame rate of video 1 and video 2 is 2 frames/sec and the frame rate of video 3 and video 4 is 2.5 frames/sec.

8.5.2 Task 1

The first task is to track one moving human (i.e., target) by a moving robot in scene 1, in which the background shares some features with the tracked target. The objective of this task is to show the adaptivity of the proposed tracking method in the sense that it can adaptively track the object by automatically selecting a discriminative feature. The task-relevance of each feature dimension, i.e., $\mu_f^s/(1+\sigma_f^s)$ where $f \in \{int, rg, by, o_{0^\circ}, o_{45^\circ}, o_{90^\circ}, o_{135^\circ}, ct\}$, obtained from the online learned low-level LTM object representation of the target in scene 1, is shown in Figure 8.2. It indicates that contour is the task-relevant feature. The tracking results of the proposed method are shown in Figure 8.3(m) - 8.3(p): The proposed method succeeds in tracking the target when it is passing by the red board. Results of the Camshift algorithm are shown in Figure 8.3(q) - 8.3(t): It fails to track the target when it is passing by the red board, since the red board shares hue values with the target.

8.5.3 Task 2

The second task is to track one moving human (i.e., target) by a moving robot in scene 2, in which there is full occlusion during several sequential frames. The objective of this task is to show that the proposed method can automatically recover the target after it goes through the full occlusion. The task-relevance of each feature dimension, i.e., $\mu_f^s/(1+\sigma_f^s)$ where $f \in \{int, rg, by, o_{0^\circ}, o_{45^\circ}, o_{90^\circ}, o_{135^\circ}, ct\}$, obtained from the online learned low-level LTM object representation of the target in scene 2, is shown in Fig 8.4. It indicates that contour is the task-relevant feature. The tracking results of the proposed method are shown in Figure 8.5(m) - 8.5(p): The proposed method succeeds in tracking the target after it goes through the full occlusion. Results of the Camshift algorithm are shown in Figure 8.5(q) - 8.5(t): The tracking region covers almost the whole scene after the target goes through the occlusion, so the CamShift algorithm fails to recover the tracking after the full occlusion.

8.5.4 Task 3

The third task is to track one moving human (i.e., target) by a moving robot in scene 3 in which there is another moving robot (i.e., distractor). The objective of this task is to show that the proposed tracking method is effective in the environment with distractors and clutter. The task-relevance of each feature dimension, i.e., $\mu_f^s/(1+\sigma_f^s)$ where $f \in \{int, rg, by, o_{0^\circ}, o_{45^\circ}, o_{90^\circ}, o_{135^\circ}, ct\}$, obtained from the online learned low-level LTM object representation of the target in scene 3, is shown in Figure 8.6. It indicates that contour is the task-relevant feature. As the target is small in this video, the head part of the target human is not segmented. As a result, the learned target has only one part. The tracking results of the proposed method are shown in Figure 8.7(m) - 8.7(p): The proposed method succeeds in tracking the target. Results of the Camshift algorithm are shown in Figure 8.7(q) - 8.7(t): It fails to track the target when the target passes by the dark door (Figure 8.7(t)).

8.5.5 Task 4

The fourth task is to track one moving human (i.e., target) by a moving robot in scene 4 in which there is another moving human (i.e., distractor). One objective of this task is to show that the proposed method is robust to variations of lighting on the target. The other objective is to show the proposed method can provide the completion of the tracked target that includes several parts. The task-relevance of each feature dimension, i.e., $\mu_f^s/(1+\sigma_f^s)$ where $f \in \{int, rg, by, o_{0^\circ}, o_{45^\circ}, o_{90^\circ}, o_{135^\circ}, ct\}$, obtained from the online learned low-level LTM object representation of the target in scene 4, is shown in Figure 8.8. It shows that red-green of the part 2 (i.e., the upper body of the target) is the task-relevant feature. The tracking results of the proposed method are shown in Figure 8.9(m) - 8.9(p): The proposed method succeeds in tracking the target and achieves target completion. Results of the Camshift algorithm are shown in Figure 8.9(q) - 8.9(t): It fails to track the target when the target passes by the blue door (Figure 8.9(t)).

8.5.6 Performance Evaluation

Tracking performance is evaluated by using tracking precision P_{TPR}, which is calculated as a true positive rate:

$$P_{TPR} = nTP/nTOT, \tag{8.1}$$

where nTP is the number of frames in which the target is correctly detected and $nTOT$ is the total number of frames in a video.

Target completion is evaluated by using both true positive rate C_{TPR} and false positive rate C_{FPR}, which are calculated respectively as:

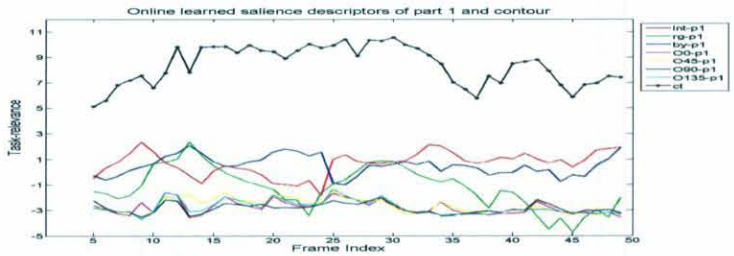
$$\begin{aligned} C_{TPR} &= A_{TP}/A_{real} \\ C_{FPR} &= A_{FP}/A_{real} \end{aligned}, \tag{8.2}$$

where A_{real} is the pixel number of the real target, A_{TP} is the number of pixels that are both in the tracked region and in the real target, and A_{FP} is the number of pixels that are in the tracked region but not in the real target. Note that target completion is evaluated only for frames in which the target is tracked successfully.

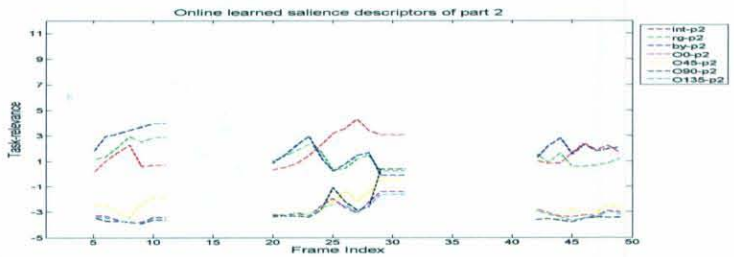
Performance evaluation of the proposed method and the Camshift algorithm is shown in Table 8.1. It can be seen that the tracking performance and target completion performance in the proposed method are both better than those in the CamShift algorithm. In task 2, P_{TPR} is decreased in the proposed method since the target is fully occluded in a total of 15 sequential frames. Except for the frames of full occlusion, the proposed method can successfully track the target in different scenes and viewing conditions and can achieve the precise target region, shape and size as well.

8.6 Conclusion

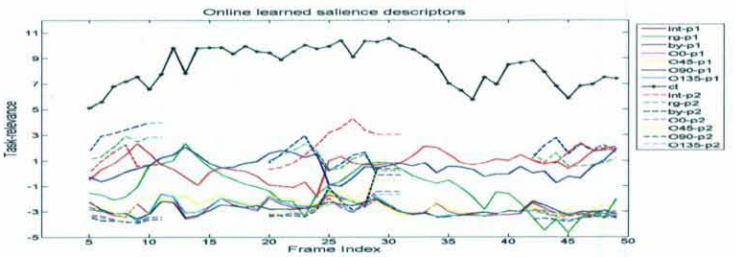
This chapter has presented a target tracking method using the proposed cognitive visual perception paradigm. This tracking method consists of four modules: online learning



(a)



(b)



(c)

Figure 8.2: Learning results of task 1: The task-relevance of each feature dimension obtained from the online learned low-level LTM object representation of the target in scene 1. (a) Part 1 (Body of the human) and global contour. (b) Part 2 (Head of the human). This part is invisible in several frames due to the target's posture. (c) Combination of part 1, part 2 and global contour.

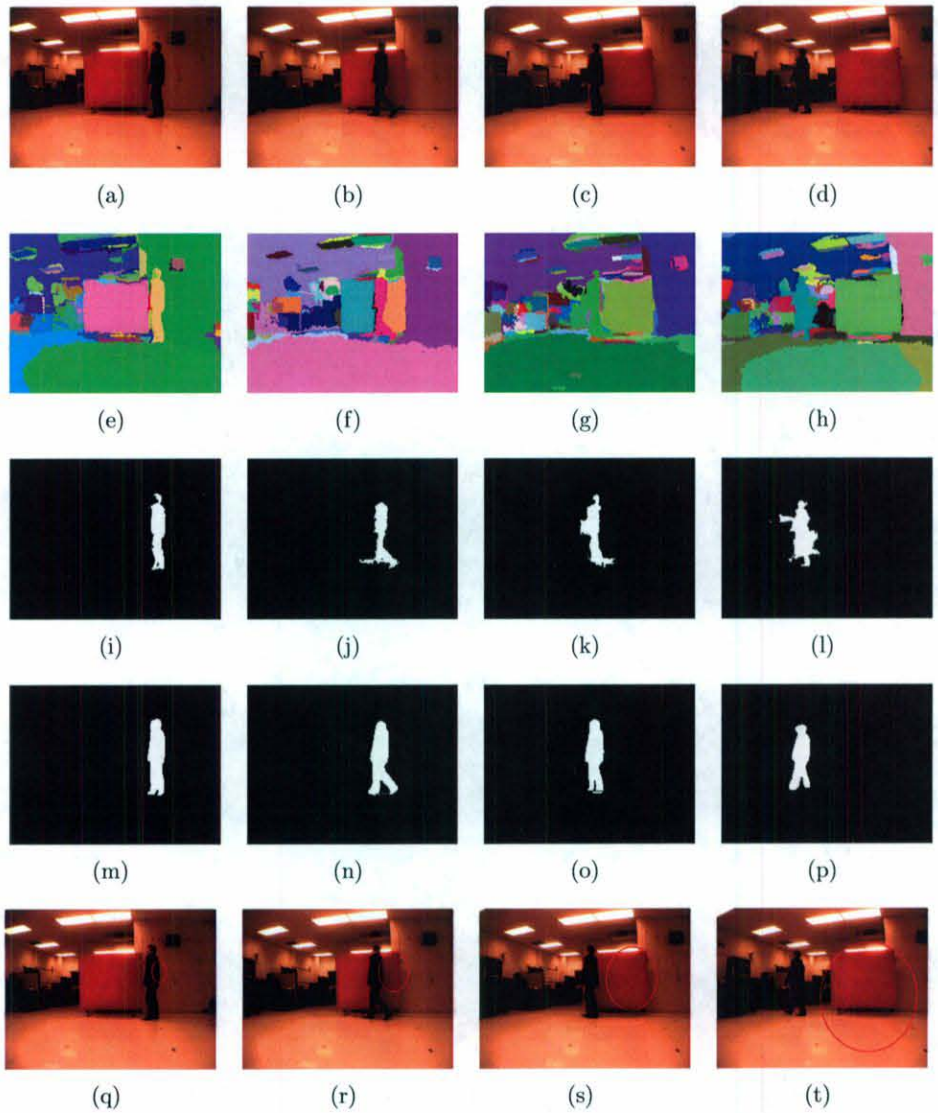
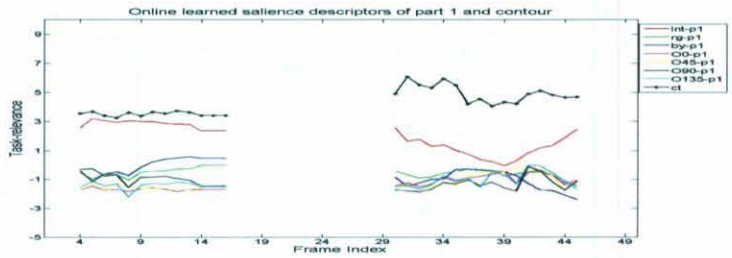
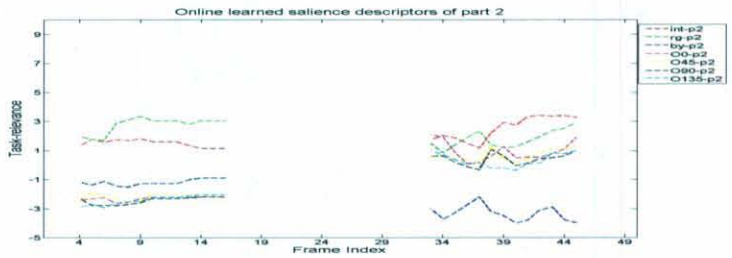


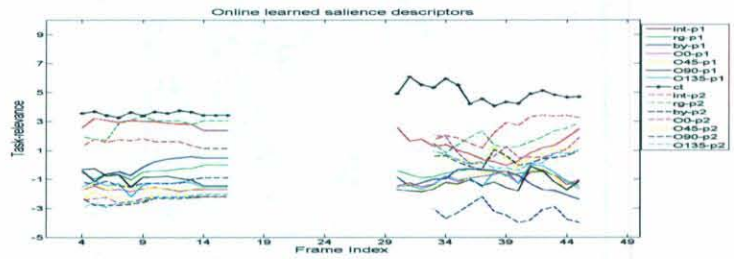
Figure 8.3: Tracking results of task 1: Tracking of a moving human by the moving robot in scene 1, in which the background shares some features with the target. (a)-(d) Original images in frame 9, 11, 13 and 16 from video 1. (e)-(h) Pre-attentive segmentation. Each color represents a proto-object. (i)-(l) Proto-object based attentional activation map. Brightness represents attentional activation. (m)-(p) The final tracking region after post-attentive completion processing. (q)-(t) Tracking results using the CamShift algorithm. Red ellipses represent the tracking regions.



(a)



(b)



(c)

Figure 8.4: Learning results of task 2: The task-relevance of each feature dimension obtained from the online learned low-level LTM object representation of the target in scene 2. (a) Part 1 (Body of the human) and global contour. Part 1 is invisible in several frames due to the full occlusion. (b) Part 2 (Head of the human). Part 2 is invisible in several frames due to the full occlusion and target's posture. (c) Combination of part 1, part 2 and global contour.

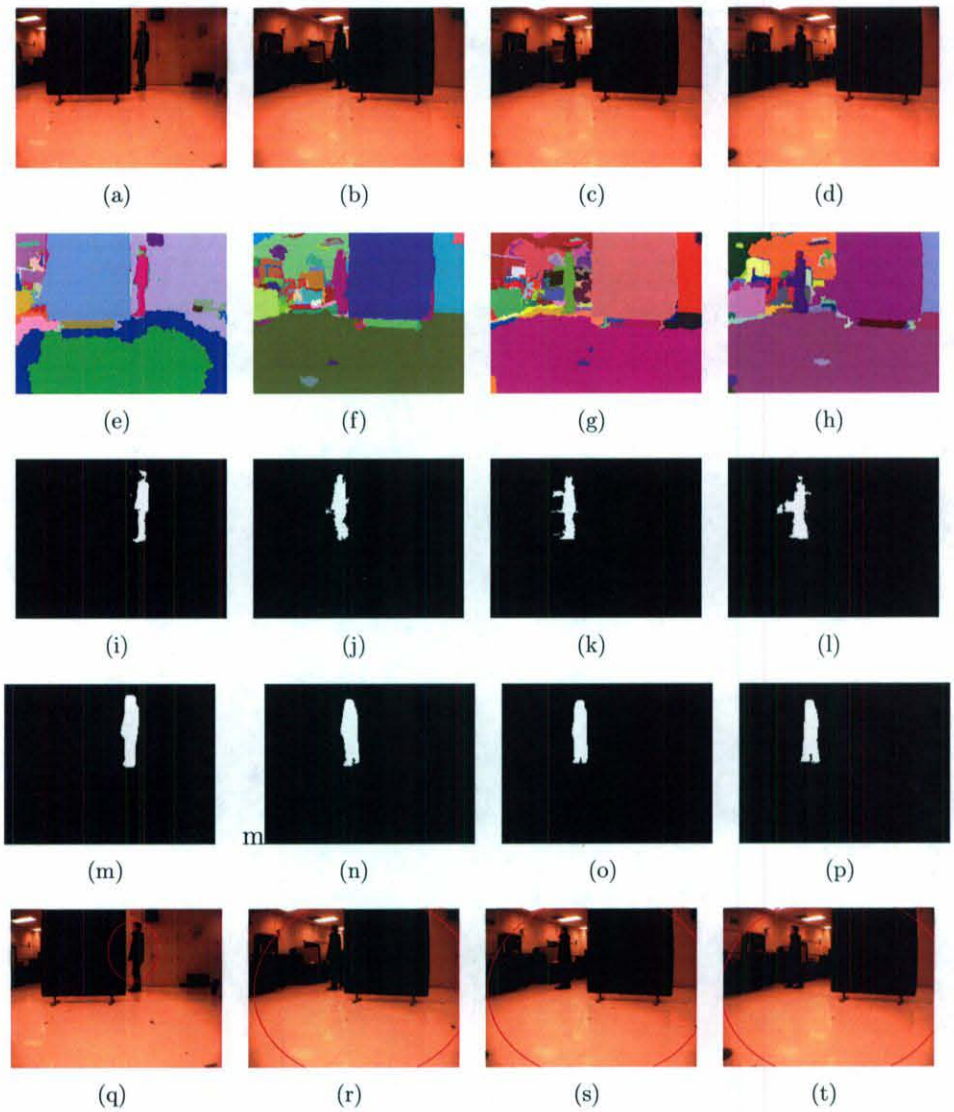


Figure 8.5: Tracking results of task 2: Tracking of a moving human by the moving robot in scene 2, in which full occlusion exists. (a)-(d) Original images in frame 13, 31, 33 and 35 from video 2. (e)-(h) Pre-attentive segmentation. Each color represents a proto-object. (i)-(l) Proto-object based attentional activation map. Brightness represents attentional activation. (m)-(p) The final tracking region after post-attentive completion processing. (q)-(t) Tracking results using the CamShift algorithm. Red ellipses represent the tracking regions.

Table 8.1: Tracking performance. In this table, "T" means task, "M" means method, "O" means the proposed method and "C" means the Camshift method.

T	M	Frm #	P_{TPR} (%)	C_{TPR} (%)	C_{FPR} (%)
1	O	44	100.00	92.71	6.70
	C	44	11.36	39.34	2.09
2	O	42	64.29	91.60	8.13
	C	42	26.19	36.38	3.32
3	O	43	100.00	95.15	5.31
	C	43	86.05	93.06	34.83
4	O	65	96.92	97.80	2.48
	C	65	80.00	92.50	5.09

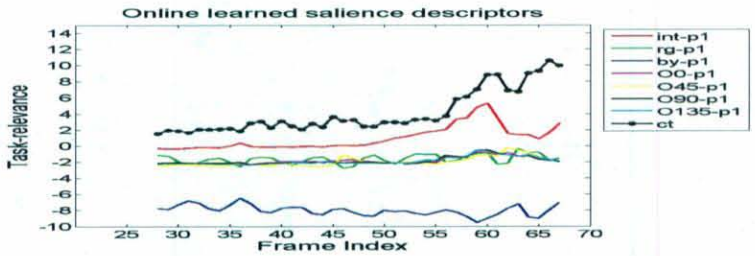


Figure 8.6: Learning results of task 3: The task-relevance of each feature dimension obtained from the online learned low-level LTM object representation of the target in scene 3.

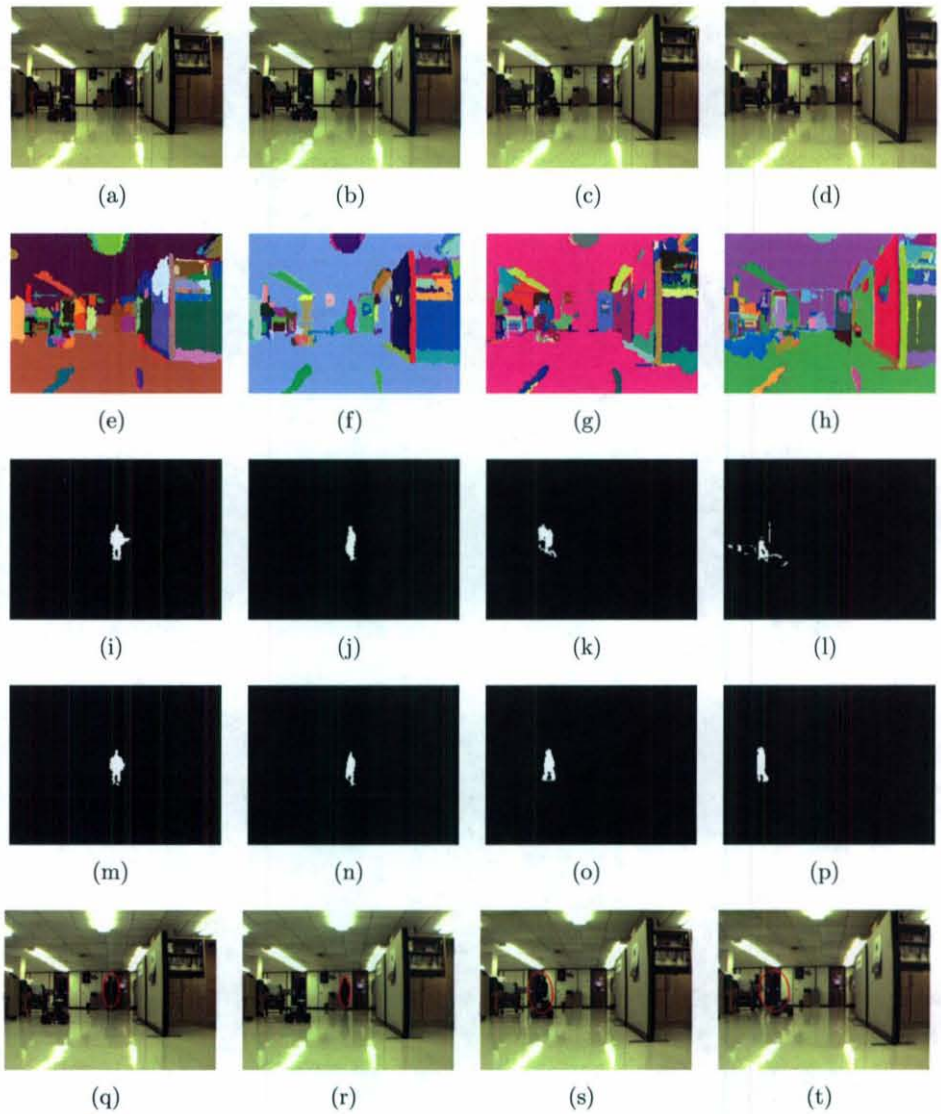
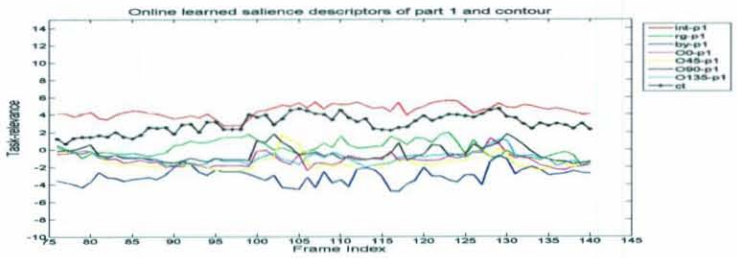
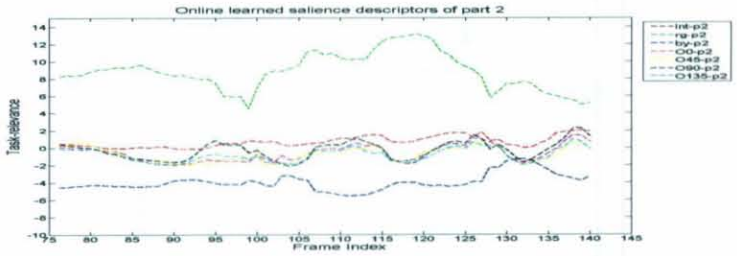


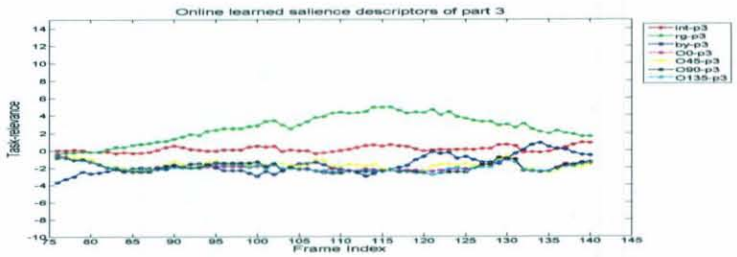
Figure 8.7: Tracking results of task 3: Tracking of a moving human by the moving robot in scene 3, in which there is another moving robot. (a)-(d) Original images in frame 46, 51, 61 and 66 from video 3. (e)-(h) Pre-attentive segmentation. Each color represents a proto-object. (i)-(l) Proto-object based attentional activation map. Brightness represents attentional activation. (m)-(p) The final tracking region after post-attentive completion processing. (q)-(t) Tracking results using the CamShift algorithm. Red ellipses represent the tracking regions.



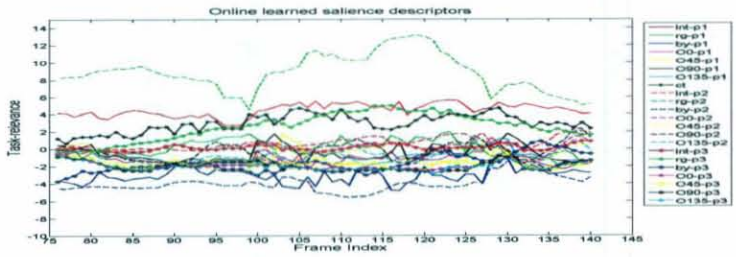
(a)



(b)



(c)



(d)

Figure 8.8: Learning result of task 4: The task-relevance of each feature dimension obtained from the online learned low-level LTM object representation of the target in scene 4. (a) Part 1 (Head of the human) and global contour. (b) Part 2 (Upper body part of the human). (c) Part 3 (Lower body part of the human). (d) Combination of part 1, part 2, part 3 and global contour.

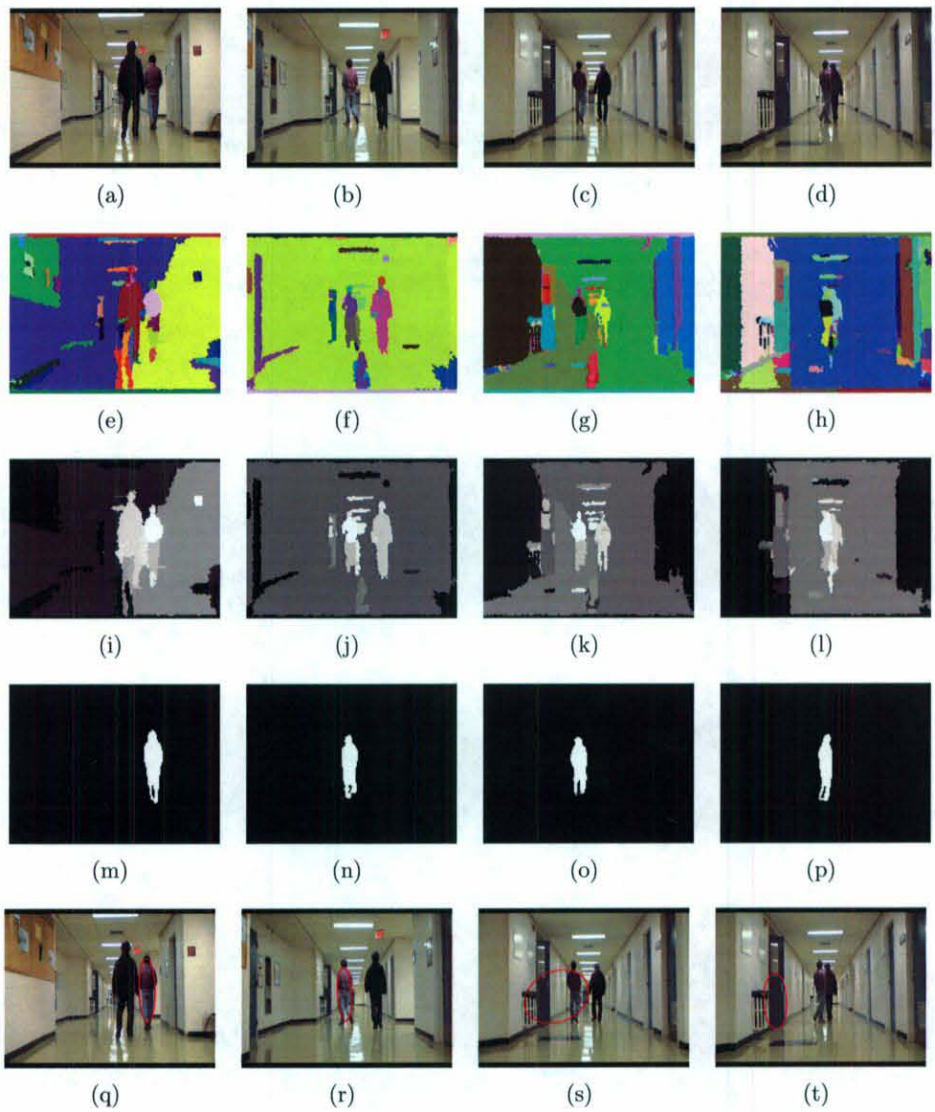


Figure 8.9: Tracking results of task 4: Tracking of a moving human by the moving robot in scene 4, in which another moving human exists and the lighting conditions on the target is changing. (a)-(d) Original images in frame 90, 103, 126 and 128 from video 4. (e)-(h) Pre-attentive segmentation. Each color represents a proto-object. (i)-(l) Proto-object based attentional activation map. Brightness represents attentional activation. (m)-(p) The final tracking region after post-attentive completion processing. (q)-(t) Tracking results using the CamShift algorithm. Red ellipses represent the tracking regions.

of the target model, pre-attentive segmentation, attentional selection and post-attentive recognition. Compared with other tracking methods, this proposed method has three advantages. The first one is adaptivity. The task-relevant feature(s), which can globally discriminate the target and the background, can be autonomously deduced online from the learned salience descriptors such that they can cope with cluttered and dynamically changing environments. Furthermore, a broad feature space can be used as the candidate of the task-relevant feature(s). The second one is robustness. It has the ability to automatically recover tracking failure caused by any reason. The last one is precision and completion of the tracked target. By the combination of pre-attentive segmentation and post-attentive completion processing, the precise and complete target region is achieved. Experimental results in natural and cluttered scenes have shown that this proposed tracking method can achieve satisfactory tracking performance and it is capable of coping with the difficulties including appearance changes of the background and the target, large variation of motion, partial and full occlusion and so on.

Chapter 9

Conclusions and Future Perspectives

The main focus of this thesis is to investigate the visual attention mechanism and outline a cognitive visual perception paradigm for robots by using the object-based visual attention mechanism. The rest of this chapter will summarize the research issues that have been addressed in this thesis and present future research directions.

9.1 Research Summary

Robots produce corresponding actions based on perceptual information obtained from a variety of sensing systems, among which vision is one of the primary modalities. Unlike traditional robots whose perceptual behaviors are manually designed by programmers for a given task, truly intelligent robots should have the mental capability of knowing how to perceive the environment autonomously. Based on the psychological and physiological fact that humans employ a visual attention mechanism to connect perception and action in the sense that only the relevant parts of the environment are selected to be present for actions, this thesis has presented a cognitive visual perception paradigm that determines how visual inputs reach awareness and guide actions. This thesis further asserts that two aspects are required for the cognitive visual perception system. One is the conscious aspect that can direct perception based on the task, context and knowledge learned from

experience. The other is the unconscious aspect that can direct perception in the case of facing an unexpected, unusual or surprise situation.

The proposed paradigm divides visual perception into three successive stages: pre-attentive processing, attentional selection and post-attentive perception. Using this paradigm, robotic visual perception starts from a low-level cognitive attentional selection procedure that guides attention to an object of the scene, followed by a high-level post-attentive analysis procedure that analyzes the attended object and formulates it into an internal mental representation used for further cognitive behaviors.

The pre-attentive processing stage extracts low-level pre-attentive features and then segments the input scene into homogeneous proto-objects in a bottom-up, unsupervised fashion. The contribution of this stage is the pre-attentive segmentation algorithm. It is based on the irregular pyramid techniques and has several innovative extensions, including a scale-invariant probabilistic similarity measure, a data-driven pyramidal decimation method and a similarity-based neighbor search method. Experimental results have shown that the proposed pre-attentive segmentation algorithm provides satisfactory results.

The attentional selection stage involves four modules: bottom-up competition, top-down biasing, the combination of bottom-up saliency and top-down biases, and the estimation of proto-object based attentional activation. The bottom-up competition module aims to model the unconscious aspect of visual perception and generates a location-based bottom-up saliency map.

The top-down biasing module aims to model the conscious aspect of visual perception based on Duncan's IC hypothesis [49] and generates a location-based top-down bias map. The top-down biasing method is one contribution of the attentional selection stage. In this method, a task-relevant object is first deduced from the task, then one or a few task-relevant feature(s) are deduced from the LTM representation of the task-relevant object, and finally top-down biases in terms of the task-relevant feature dimension(s) are estimated by using a Bayesian inference process. Thus, this top-down biasing method has the following four advantages: effectiveness, efficiency, adaptability and robustness.

After bottom-up competition and top-down biasing, a location-based attentional activation map is obtained by combining the bottom-up saliency map and top-down bias map at a unified probabilistic scale. Finally, a proto-object based attentional activation map is obtained by combining the activation contributions within each proto-object.

Following the attentional selection stage, the attended proto-object proceeds to the post-attentive perception stage, which includes four functional modules: perceptual completion processing, extraction of post-attentive features, development of LTM object representations and object recognition. The main function of the post-attentive perception stage is to interpret the attended object in detail to produce an appropriate action at the current moment, to update the corresponding LTM object representation at the current moment, and to consciously guide the top-down biasing at the next moment. The main contribution of this stage is the PNN based LTM object representation. One advantage of this proposed LTM object representation is that it can probabilistically embody various instances of that object. The other advantage is that it includes two probabilistic combination methods (i.e., probabilistic mixture and probabilistic summary) so that it can be used for both high-level post-attentive analysis and low-level top-down biasing. The result is that the learned LTM representation is robust and discriminative. Dynamical learning algorithms are also developed for training the PNN based object representations.

The proposed cognitive visual perception paradigm has been applied to two types of robotic tasks. The first type of task is object detection, including the detection of salient objects using the bottom-up attention mechanism and the detection of task-specified targets using the top-down attention mechanism. The second type of task is target tracking. The processes of detection and tracking both include the attentional selection module and the post-attentive recognition module. The attentional selection module can rapidly localize a candidate object by using either bottom-up attention or top-down attention. The following post-attentive recognition module can validate the attended object by using high-level analysis. Experimental results have shown that the proposed perception paradigm can achieve a satisfactory detection and tracking performance to

cope with difficulties, including changes in the background and the target, large variation of motion, partial and full occlusion and so on.

9.2 Publications Related to the Research Work

This thesis is based on the following technical publications that report the contributions of the proposed work.

1. Yuanlong Yu, George K. I. Mann, and Raymond G. Gosine, "An Object-based Visual Attention Model for Robotic Applications", in *IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics*, appearing in 2010.

2. Yuanlong Yu, George K. I. Mann, and Raymond G. Gosine, "Target Tracking for Moving Robots Using Object-based Visual Attention", in the *Proceedings of IEEE International Conference on Robotics and Automation*, Taibei, Taiwan, October, 2010.

3. Yuanlong Yu, George K. I. Mann, and Raymond G. Gosine, "A Novel Robotic Visual Perception Method Using Object-based Attention", in the *Proceedings of IEEE International Conference on Robotics and Biomimetics*, Guilin, China, December, 2009.

4. Yuanlong Yu, George K. I. Mann, and Raymond G. Gosine, "An Autonomous Visual Perception Model for Robots Using Object-based Attention Mechanism", in the *Proceedings of IEEE International Conference on Robotics and Biomimetics*, Guilin, China, December, 2009.

5. Yuanlong Yu, George K. I. Mann, and Raymond G. Gosine, "Modeling of Top-down Influences on Object-based Visual Attention for Robots", in the *Proceedings of IEEE International Conference on Robotics and Biomimetics*, Guilin, China, December, 2009.

6. Yuanlong Yu, George K. I. Mann, and Raymond G. Gosine, "Modeling of Top-down Object-based Attention Using Probabilistic Neural Network", in the *Proceedings of IEEE Canadian Conference on Electrical and Computer Engineering*, St. John's, Canada, May, 2009.

7. Yuanlong Yu, George K. I. Mann, and Raymond G. Gosine, "An Object-based Visual Attention Model for Robots", in the *Proceedings of IEEE International Conference on Robotics and Automation*, Pasadena, California, USA, May 2008.

8. Yuanlong Yu, George K. I. Mann, and Raymond G. Gosine, "A Task-driven Object-based Attention Model for Robots", in the *Proceedings of IEEE International Conference on Robotics and Biomimetics*, Sanya, China, December 2007.

9. Yuanlong Yu, George K. I. Mann, and Raymond G. Gosine, "Task-driven Moving Object Detection for Robots Using Visual Attention", in the *Proceedings of IEEE-RAS International Conference on Humanoid Robots*, Pittsburgh, USA, November 2007.

9.3 Future Research Directions

The proposed cognitive visual perception paradigm leads to several potential research topics in the area of cognitive robotics.

An important potential research topic is cognitive perception-action mapping. Cognitive perception-action mapping can be generally defined as an association between perception, context and actions. According to the proposed cognitive visual perception paradigm, this cognitive mapping can be modeled as an association between attentional states, context and actions. Actions of a cognitive robot can be categorized into two types. The first type is external actions, which guide the operation of effectors. The second type is internal actions, which mainly includes guidance for attentional selection at the next moment. Thus, cognitive perception-action mapping can be modeled to fulfill

two functions. The first function is the association between the current attentional state and the current action, termed as *mapping between attention and external actions*. In other words, the attended object is recognized by finding a matched instance in LTM and then the matched instance is used to select an appropriate action based on the learned perception-action mapping. The second function is the association between the current attentional state and the next possible attentional state (i.e., attentional prediction), termed as *mapping between attention and internal actions*. Since the proposed cognitive perception paradigm is object-based, the attentional state is an instance of the object that is attended at the current moment and attentional prediction is an instance of the task-relevant object at the next moment.

A potential research approach to modeling cognitive perception-action mapping is the FDMP. An FDMP of perception-action mapping can be expressed as $p(act_t^e, act_{t+1}^i | attn_t)$, where $attn_t$ denotes the attentional state at moment t, act_{t+1}^e denotes the external action at moment t, and act_{t+1}^i denotes the attentional prediction at moment $t+1$. This definition means that the probability of each candidate action and each candidate attentional prediction can be estimated given the attentional state at the current moment. According to the proposed object-based cognitive perception paradigm, the set of discrete attentional states is composed of the developed LTM object representations.

Another potential research topic is the integration of the surprise mechanism into the attentional selection stage in the proposed cognitive visual perception paradigm. Surprise is a mechanism that can attract attention to an unusual or an unexpected item in the temporal context. In other words, it is referred to as temporal novelty. The integration of surprise can enable robots to perceive novel objects and events in an unconscious manner.

Appendix A

Gaussian Pyramid

The Gaussian pyramid technique [121] includes two types of operations: generation and interpolation.

A.1 Gaussian Pyramid Generation

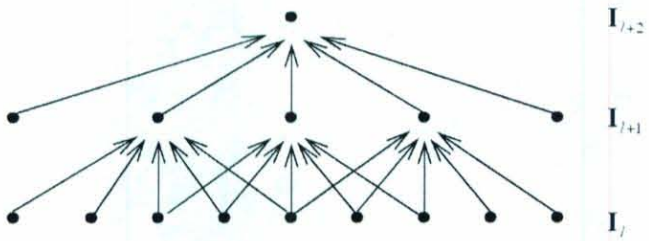


Figure A.1: Graphic representation of the generation operation of a 1-D Gaussian pyramid.

A Gaussian pyramid is a sequence of images $\mathbf{I}_0, \mathbf{I}_1, ..., \mathbf{I}_n$ in order of both decreased resolution and sample density. A one-dimensional graphic representation of the generation operation of the Gaussian pyramid is given in Figure A.1. The purpose of the Gaussian pyramid is to progressively low-pass filter and sub-sample the image on the original scale. Thus the Gaussian pyramid can be obtained by convolving the image on the original scale with one of local, symmetric weighting functions. The convolution

procedure used in this thesis can be shown as:

$$I_l(i,j) = \sum_{m=-2}^{2} \sum_{n=-2}^{2} w(m,n) I_{l-1}(2i+m, 2j+n), \tag{A.1}$$

where I_l denotes the convoluted image at scale $l > 0$, (i,j) represents the spatial coordinates of a point in I_l, I_0 is the input image at the original scale, and $w(m,n)$ is a two-dimensional 5-by-5 weighting function that should satisfy three constraints. These three constraints for a one-dimensional weight function $w(m)$ can be expressed as:

$$\begin{aligned} w(0) &= c, \\ w(-1) &= w(1) = 1/4, \\ w(-2) &= w(2) = 1/4 - c/2, \end{aligned} \tag{A.2}$$

where c is the parameter used to determine the shape of the weighting function. In the case $c = 0.4$, the shape of the weighting function is Gauss-like.

Based on the 1-D weighting function, the 2-D weighting function can be calculated as:

$$w(m,n) = w(m)w(n). \tag{A.3}$$

A.2 Gaussian Pyramid Interpolation

Interpolation is a reverse convolution operation of the Gaussian pyramid. Its effect is to expand a low-resolution image into a high-resolution image by interpolating new nodes between the given ones. The interpolation procedure can be expressed as:

$$I_{(l)to(l-1)}(i,j) = 4 \sum_{m=-2}^{2} \sum_{n=-2}^{2} w(m,n) I_l(\frac{i-m}{2}, \frac{j-n}{2}). \tag{A.4}$$

Appendix B

2-D Gabor Filters

The 2-D Gabor filter is the product of a complex sinusoidal, known as the carrier, and a Gaussian-shaped function, known as the envelope. As shown in [122], a family of 2-D Gabor functions and their Fourier transforms can be respectively expressed as :

$$g_\theta(x,y) = K\exp\{-\pi[(x-x_0)^2a^2+(y-y_0)^2b^2]\}\times \exp\{-2\pi i[u_0(x-x_0)+v_0(y-y_0)]\} \quad (B.1)$$

$$G_\theta(u,v) = K\exp\{-\pi[(u-u_0)^2/a^2+(v-v_0)^2/b^2]\}\times \exp\{-2\pi i[x_0(u-u_0)+y_0(v-v_0)]\} \quad (B.2)$$

where (u_0, v_0) are the spatial center frequencies where the filter has the maximal responses and those two center frequencies determine the orientation θ, (a, b) are standard deviations of the Gaussian envelope and determine the bandwidth of the filter, (x_0, y_0) are the centroid of the Gaussian envelope in the space domain and they are set as $(0, 0)$ in this thesis, and K scales the magnitude of the Gaussian envelope.

Since actual impulse-response functions and neural receptive field profiles are real functions, the above Gabor filter can be decomposed into a quadrature form including an even-symmetry (cosine) part as shown in (B.3) and an odd-symmetry (sine) part as

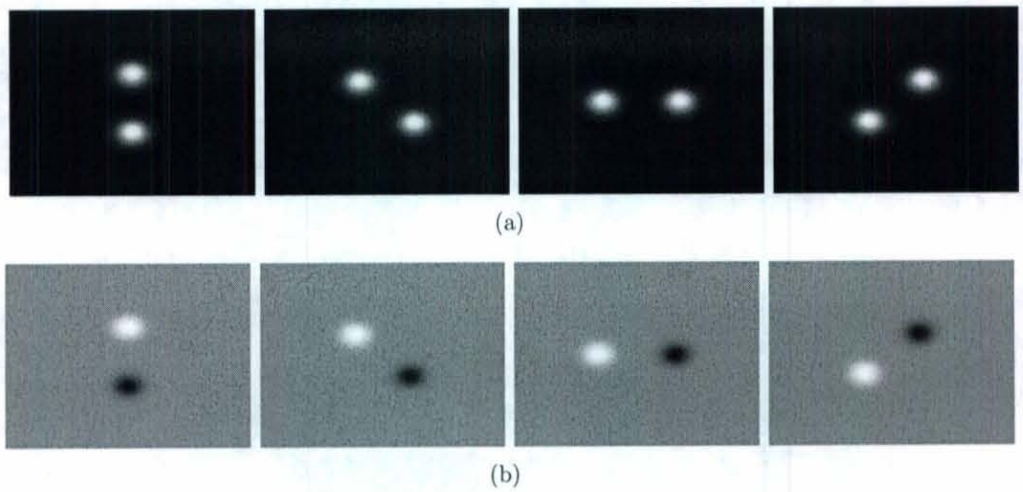


Figure B.1: Examples of 2-D Gabor filters (in the frequency domain) in four orientations. Column 1: In orientation $\theta = 0°$. Column 2: In orientation $\theta = 45°$. Column 3: In orientation $\theta = 90°$. Column 4: In orientation $\theta = 135°$. Row 1: Even-symmetry parts. Row 2: Odd-symmetry parts.

shown in (B.4):

$$g_{\theta,0}(x,y) = K \exp\{-\pi[(x-x_0)^2a^2 + (y-y_0)^2b^2]\} \times \cos\{-2\pi[u_0(x-x_0) + v_0(y-y_0)]\}, \tag{B.3}$$

$$g_{\theta,-\frac{1}{2}\pi}(x,y) = iK \exp\{-\pi[(x-x_0)^2a^2 + (y-y_0)^2b^2]\} \times \sin\{-2\pi[u_0(x-x_0) + v_0(y-y_0)]\}. \tag{B.4}$$

The corresponding forms in the frequency domain can be respectively expressed as:

$$G_{\theta,0}(u,v) = 1/2[G_\theta(u,v) + G_\theta(-u,-v)], \tag{B.5}$$

$$G_{\theta,-\frac{1}{2}\pi}(u,v) = 1/2[-iG_\theta(u,v) + iG_\theta(-u,-v)]. \tag{B.6}$$

The examples of even-symmetry parts and odd-symmetry parts of the Gabor filters in four preferred orientations (i.e., $\theta \in \{0°, 45°, 90°, 135°\}$) are shown in Figure B.1.

In order to extract the orientation energy at multiple scales, a set of multi-scale 2-D Gabor filters is used to convolve the intensity images at the corresponding scales. These multi-scale 2-D Gabor filters are obtained by adjusting the parameters (a, b, u_0, v_0) for each scale.

Given the intensity image $\mathbf{F}_{int}$ at scale l, it is convolved with the even-symmetric part and the odd-symmetric part respectively of the 2-D Gabor filter at scale l. The orientation energy in a preferred orientation θ at scale l can be finally obtained by a combination of the convolution results of the even-symmetric part and the odd-symmetric part:

$$F_{o_\theta}(\mathbf{r}_i, l) = \sqrt{r^2_{\theta,0}(\mathbf{r}_i, l) + r^2_{\theta,-\frac{1}{2}\pi}(\mathbf{r}_i, l)}, \tag{B.7}$$

where $r_{\theta,0}(\mathbf{r}_i, l)$ and $r_{\theta,-\frac{1}{2}\pi}(\mathbf{r}_i, l)$ respectively represents the convolution results of the even-symmetric part and the odd-symmetric part of the 2-D Gabor filter at a pixel $\mathbf{r}_i$ at scale l.

Bibliography

[1] S. Russell and P. Norvig, *Artificial intelligence: A model aprroach.* Prentice-Hall, Upper Saddle River, NJ, 1995, pp. 31–52.

[2] R. C. Gonzalez and R. E. Woods, *Digital image processing*, 2nd ed. Prentice Hall, Upper Saddle River, NJ, 2001, pp. 134–137.

[3] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679–698, 1986.

[4] N. Aggarwal and W. C. Karl, "Line detection in images through regularized hough transform," *IEEE Transactions on Image Processing*, vol. 15, no. 3, pp. 582–591, 2006.

[5] C. Harris and M. J. Stephens, "A combined corner and edge detector," in *Proceedings of the 4th Alvey Vision Conference, Manchester, UK*, 1988, pp. 147–151.

[6] K. Sohn, J. H. Kim, and W. E. Alexander, "A mean field annealing approach to robust corner detection," *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics*, vol. 28, no. 1, pp. 82–90, 1998.

[7] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the International Conference on Computer Vision (ICCV), Corfu*, 1999, pp. 1150–1157.

[8] J. Wang, H. Zha, and R. Cipolla, "Coarse-to-fine vision-based localization by indexing scale-invariant features," *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics*, vol. 36, no. 2, pp. 413–422, 2006.

[9] L. Armesto, G. Ippoliti, S. Longhi, and J. Tornero, "Probabilistic self-localization and mapping - an asynchronous multirate approach," *IEEE Robotics and Automation Magazine*, vol. 15, no. 23, pp. 77–88, 2008.

[10] A. A. Argyros, K. E. Bekris, and S. C. Orphanoudakis, "Robot homing based on corner tracking in a sequence of panoramic images," in *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2001, pp. 3–10.

[11] H. Katsura, J. Miura, M. Hild, and Y. Shirai, "A view-based outdoor navigation using object recognition robust to changes of weather and seasons," in *Proceedings of IEEE/RSJ International Conference of Inteligent Robots and Systems (IROS)*, 2003, pp. 2974–2979.

[12] Y. Matsumoto, M. Inaba, and H. Inoue, "View-based approach to robot navigation," in *Proceedings of IEEE/RSJ International Conference of Inteligent Robots and Systems (IROS)*, 2000, pp. 1702–1708.

[13] R. Murrieta, C. Parra, and M. Devy, "Visual navigation in natural environments: From range and color data to a landmark based model," *Autonomous Robots*, vol. 13, no. 2, pp. 143–168, 2002.

[14] I. Ulrich and I. Nourbakhsh, "Appearance-based place recognition for topological localization," in *Proceedings of the IEEE International Conference of Robotics and Automation (ICRA)*, 2000, pp. 1023–1029.

[15] L. Renniger and J. Malik, "When is scene identification just texture recognition?" *Visual Research*, vol. 44, no. 19, pp. 2301–2311, 2004.

[16] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International Journal of Computer Vision*, vol. 42, no. 3, pp. 145–175, 2001.

[17] C. Siagian and L. Itti, "Biologically-inspired robotics vision monte-carlo localization in the outdoor environment," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2007, pp. 1723–1730.

[18] ——, "Biologically inspired mobile robot vision localization," *IEEE Transactions on Robotics*, vol. 25, no. 4, pp. 861–873, 2009.

[19] A. Remazeilles and F. Chaumette, "Image-based robot navigation from an image memory," *Robotics and Autonomous Systems*, vol. 55, no. 4, pp. 345–356, 2007.

[20] R. J. Brachman, "Systems that know what they're doing," *IEEE Intelligent Systems*, vol. 17, no. 6, pp. 67–71, 2002.

[21] D. A. Norman, "Toward a theory of memory and attention," *Psychological Review*, vol. 75, no. 6, pp. 522–536, 1968.

[22] D. E. Broadbent, "Levels, hierarchies, and the locus of control," *Quarterly Journal of Experimental Psychology*, vol. 29, no. 2, pp. 181–201, 1977.

[23] M. I. Posner, "Orienting of attention," *Quarterly Journal of Experimental Psychology*, vol. 32, no. 1, pp. 3–25, 1980.

[24] A. M. Treisman and G. Gelade, "A feature integration theory of attention," *Cognition Psychology*, vol. 12, no. 1–2, pp. 507–545, 1980.

[25] J. Duncan, "Selective attention and the organization of visual information," *Journal of Experimental Psychology: General*, vol. 113, no. 4, pp. 501–517, 1984.

[26] S. P. Tipper, L. A. Howard, and G. Houghton, "Action-based mechanisms of attention," *Philosophical Transactions: Biological Sciences*, vol. 353, no. 1373, pp. 1385–1393, 1998.

[27] E. k. Miller, "Prefrontal cortex and the neural basis of executive functions," in *Attention, Space and Action: Studies in Cognitive Neuroscience*, G. W. Humphreys, J. Duncan, and A. Treisman, Eds. Oxford University Press, 1999, pp. 251–272.

[28] M. I. Posner, C. R. R. Snyder, and B. J. Davidson, "Attention and the detection of signals," *Journal of Experimental Psychology: General*, vol. 14, no. 2, pp. 160–174, 1980.

[29] C. W. Eriksen and J. D. S. James, "Visual attention within and around the field of focal attention: A zoom lens model," *Perception and Psychophysics*, vol. 40, no. 4, pp. 225–240, 1986.

[30] C. J. Downing, "Expectancy and visual-spatial attention: Effects on perceptual quality," *Journal of Experimental Psychology: Human Percetion and Performance*, vol. 14, no. 2, pp. 188–202, 1988.

[31] N. Kanwisher and J. Driver, "Objects, attributes, and visual attention: Which, what, and where," *Current Directions in Psychological Science*, vol. 1, no. 1, pp. 26–31, 1992.

[32] S. P. Vecera and M. J. Farah, "Does visual attention select objects or locations?" *Journal of Experimental Psychology: General*, vol. 123, no. 2, pp. 146–160, 1994.

[33] A. F. Kramer, T. A. Weber, and S. E. Watson, "Object-based attention selection – grouped arrays or spatially invariant representations?: Comment on vecera and farah(1994)," *Journal of Experimental Psychology: General*, vol. 126, no. 1, pp. 3–13, 1997.

[34] C. R. Olson, "Object-based vision and attention in primates," *Current Opinion in Neurobiology*, vol. 11, no. 2, pp. 171–179, 2001.

[35] M. S. Worden and J. J. Foxe, "The dynamics of the spread of selective visual attention," *Proceedings of National Academy of Sciences of the United State of America*, vol. 100, no. 21, pp. 11 933–11 935, 2003.

[36] B. J. Scholl, "Objects and attention: the state of the art," *Cognition*, vol. 80, no. 1-2, pp. 1–46, 2001.

[37] C. Koch and S. Ullman, "Shifts in selective visual attention: towards the underlying neural circuitry," *Huamn Neurobiology*, vol. 4, no. 4, pp. 219–217, 1985.

[38] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 11, pp. 1254–1259, 1998.

[39] V. Navalpakkam and L. Itti, "Modeling the influence of task on attention," *Vision Research*, vol. 45, no. 2, pp. 205–231, 2005.

[40] F. H. Hamker, "Modeling attention: From computational neuroscience to computer vision," in *Attention and Performance in Computational Vision. Second International Workshop on Attention and Performance in Computer Vision (WAPCV 2004), LNCS 3368*, L. P. et al., Ed. Berlin, Heidelberg: Springer-Verlag, 2004, pp. 118–132.

[41] S. Frintrop, "Vocus: A visual attention system for object detection and goal-directed search," Ph.D. dissertation, University of Bonn, Germany, 2005.

[42] Y. Sun and R. Fisher, "Object-based visual attention for computer vision," *Artificial Intelligence*, vol. 146, no. 1, pp. 77–123, 2003.

[43] F. Orabona, G. Metta, and G. Sandini, "Object-based visual attention: a model for a behaving robot," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Patter Recognition (CVPR)*, 2005, p. 89.

[44] J. M. Wolfe, "Guided search 2.0: A revised model of visual search," *Psychonomic Bulletin and Review*, vol. 1, no. 2, pp. 202–238, 1994.

[45] ——, "Preattentive object files: Shapeless bundles of basic features," *Vision Research*, vol. 37, no. 1, pp. 25–43, 1997.

[46] R. Desimone and J. Duncan, "Neural mechanisms of selective visual attention," *Annual Reviews of Neuroscience*, vol. 18, pp. 193–222, 1995.

[47] R. Desimone, "Visual attention mediated by biased competition in extrastriate visual cortex," *Annual Reviews of Neuroscience*, vol. 353, no. 1373, pp. 1245–1255, 1998.

[48] J. Duncan, "Cooperating brain systems in selective perception and action," in *Attention and Performance XVI*, T. Inui and J. L. McClelland, Eds. MIT Press, Cambridge, MA, 1996, pp. 549–578.

[49] J. Duncan, G. Humphreys, and R. Ward, "Competitive brain activity in visual attention," *Current Opinion in Neurobiology*, vol. 7, no. 2, pp. 255–261, 1997.

[50] J. Duncan, "Converging levels of analysis in the cognitive neuroscience of visual attention," *Philophical Transactions of The Royal Society Lond B: Biological Sceinces*, vol. 353, no. 1373, pp. 1307–1317, 1998.

[51] K. M. O'Craven, P. E. Downing, and N. Kanwisher, "fmri evidence for objects as the units of attentional selection," *Nature*, vol. 401, pp. 584–587, 1999.

[52] P. Mamassian and M. L. nd L. T. Maloney, "Bayesian modeling of visual perception," in *Probabilistic Models of the Brain*, R. P. N. Rao, B. A. Olshausen, and M. S. Lewicki, Eds. The MIT Press, Cambridge, MA, 2002, pp. 13–36.

[53] D. F. Specht, "Probabilistic neural networks," *Neural Networks*, vol. 3, no. 1, pp. 109–118, 1990.

[54] A. Montanvert, P. Meer, and A. Rosenfeld, "Hierarchical image analysis using irregular tessellations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 4, pp. 307–316, 1991.

[55] E. Sharon, A. Brandt, and R. Basri, "Fast multiscale image segmentation," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2000, pp. 70–77.

[56] J. Triesch, D. H. Ballard, M. M. Hayhoe, and B. T. Sullivan, "What you see is what you need," *Journal of Vision*, vol. 3, no. 1, pp. 86–94, 2003.

[57] C. A. Rothkopf, D. H. Ballard, and M. M. Hayhoe, "Task and context determine where you look," *Journal of Vision*, vol. 7, no. 14, pp. 1–20, 2007.

[58] S. E. Palmer, *Vision science: Photons to phenomenology.* The MIT Press, Cambridge, Massachusetts, 1999.

[59] H. Deubel and W. X. Schneider, "Saccade target selection and object recognition: Evidence for a common attentional mechanism," *Vision Research*, vol. 36, no. 12, pp. 1827–1837, 1996.

[60] R. Johansson, G. Westling, A. Backstrom, and J. Flanagan, "Eye-hand coordination in object manipulation," *The Journal of Neuroscience*, vol. 21, no. 17, pp. 6917–6932, 2001.

[61] J. M. Findlay and I. D. Gilchrist, "Active vision perspective," in *Vision and Attention*, M. Jenkin and L. R. Harris, Eds. Springer Verlag, 2001, pp. 83–103.

[62] M. C. Bushnell, M. E. Goldberg, and D. L. Robinson, "Behavioral enhancement of visual responses in monkey cerebral cortex. i. modulation in posterior parietal cortex related to selective visual attention," *Journal of Neurophysiology*, vol. 46, no. 4, pp. 755–772, 1981.

[63] J. Moran and R. Desimone, "Selective attention gates visual processing in the extrastriate cortex," *Science*, vol. 229, no. 4715, pp. 782–784, 1985.

[64] B. C. Motter, "Focal attention produces spatially selective processing in visual cortical areas v1, v2, and v4 in the presence of competing stimuli," *Journal of Neurophysiology*, vol. 70, no. 3, pp. 909–919, 1993.

[65] J. H. R. Mausell, "The brain's visual world: representation of visual targets in cerebral cortex," *Science*, vol. 270, no. 5237, pp. 764–769, 1995.

[66] S. Treue and J. H. R. Maunsell, "Attentional modulation of visual motion processing in cortical areas mt and mst," *Nature*, vol. 382, pp. 539–541, 1996.

[67] S. J. Luck, L. Chelazzi, S. A. Hillyard, and R. Desimone, "Neural mechanisms of spatial selective attention in areas v1, v2, and v4 of macaque visual cortex," *Journal of Neurophysiology*, vol. 77, no. 1, pp. 24–42, 1997.

[68] R. Egly, J. Driver, and R. Rafal, "Shifting visual attention between objects and locations," *Journal of Experimental Psychology: General*, vol. 123, no. 2, pp. 161–177, 1994.

[69] S. Tipper, B. Weaver, L. Jerreat, and A. Burak, "Object-based and environment-based inhibition of return of visual attention," *Journal of Experimental Psychology: Human Perception and Performance*, vol. 20, no. 3, pp. 478–499, 1994.

[70] C. M. Moore, S. Yantis, and B. Vaughan, "Object-based visual selection: Evidence from perceptual completion," *Psychological Science*, vol. 9, no. 2, pp. 104–110, 1998.

[71] P. R. Roelfsema, V. A. F. Lamme, and H. Spekreijse, "Object-based attention in the primary visual cortex of the macaque monkey," *Nature*, vol. 395, pp. 376–381, 1998.

[72] J. H. Reynolds, T. Pasternak, and R. Desimone, "Attention increases sensitivity of v4 neurons," *Nature*, vol. 26, pp. 703–714, 2000.

[73] M. A. Schoenfeld, C. Tempelmann, A. Martinez, J. M. Hopf, C. Sattler, H. J. Heinze, and S. A. Hillyard, "Dynamics of feature binding during object-selective attention," *Proceedings of National Academy of Sciences of the United State of America*, vol. 100, no. 20, pp. 11 806–11 811, 2003.

[74] E. Spelke, "Where perceiving ends and thinking beings: the apprehension of objects in infancy," in *Perceptual Development in Infancy*, A. Yonas, Ed. Hillsdale, NJ, 1988, pp. 197–234.

[75] J. Driver, G. Davis, C. Russell, M. Turatto, and E. Freeman, "Segmentation, attention and phenomenal visual objects," *Cognition*, vol. 80, no. 1, pp. 61–95, 2001.

[76] E. K. Miller and J. D. Cohen, "An integrative theory of prefrontal cortex function," *Annual Review of Neuroscience*, vol. 24, pp. 167–202, 2001.

[77] L. Itti and P. Baldi, "A principled approach to detecting surprising events in video," in *Proceedings of IEEE Internatiional Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005, pp. 631–637.

[78] ——, "Bayesian surprise attracts human attention," *Visual Research*, vol. 49, no. 10, pp. 1295–1306, 2009.

[79] T. N. Mundhenk, W. Einhauser, and L. Itti, "Automatic computation of an image's statistical surprise predicts performance of human observers on a natural image detection task," *Visual Research*, vol. 49, no. 13, pp. 1620–1637, 2009.

[80] J. D. Cohen, G. Aston-Jones, and M. S. Gilzenrat, "A systems-level perspective on attention and cognitive control: Guided activation, adaptive gating, conflict monitoring, and exploitation versus exploration," in *Cognitive Neuroscience of Attention*, M. I. Posner, Ed. New York: Guilford Press, 2004, pp. 71–90.

[81] D. H. Hubel and T. N. Wiesel, "Functional architecture of macaque monkey visual cortex," in *Royal Society of London Proceedings Series B*, 1977, pp. 1–59.

[82] A. Dobbins, S. W. Zucker, and M. S. Cynader, "Endstopping and curvature," *Vision Research*, vol. 29, no. 10, pp. 1371–1387, 1989.

[83] R. Desimone and L. G. Ungerleider, "Neural mechanisms of visual processing in monkeys," in *Handbook of Neuropsychology*, F. Boller and J. Grafman, Eds. New York: Elsevier, 1989, vol. 2, pp. 267–269.

[84] R. V. D. Heydt, E. Peterhans, and G. Baumgartner, "Illusory contours and cortical neuron responses," *Science*, vol. 224, no. 4654, pp. 1260–1262, 1984.

[85] K. R. Gegenfurtner, "Cortical mechanism of color vision," *Nature Reviews Neuroscience*, vol. 4, pp. 563–572, 2003.

[86] R. Desimone, T. D. Albright, C. G. Gross, and C. Bruce, "Stimulus-selective properties of inferior temporal neurons in the macaque," *Journal of Neuroscience*, vol. 4, pp. 2051–2062, 1984.

[87] E. R. Kandel, J. H. Schwartz, and T. M. Jessell, *Essentials of neural science and behavior*. McGraw-Hill/Appleton Lange, 1996, pp. 365–487.

[88] J. H. R. Maunsell, "The brain's visual world: Representation of visual targets in cerebral cortex," *Science*, vol. 270, no. 5237, pp. 764–769, 1995.

[89] M. Usher and E. Niebur, "Modeling the temporal dynamics of it neurons in visual search: A mechanism for top-down selective attention," *Journal of Cognitive Neuroscience*, vol. 8, no. 4, pp. 311–327, 1996.

[90] J. D. Wallis, K. C. Anderson, and E. K. Miller, "Single neurons in prefrontal cortex encode abstract rules," *Nature*, vol. 411, pp. 953–956, 2001.

[91] P. S. Goldman-Rakic, "Circuitry of primate prefrontal cortex and the regulation of behavior by representational memory," in *Handbook of Physiology: The Nervous System*, F. Plum, Ed. Bethesda: American Physiological Society, 1987, pp. 373–417.

[92] G. Rainer, W. F. Asaad, and E. K. Miller, "Selective representation of relevant information by neurons in the primate prefrontal cortex," *Nature*, vol. 393, pp. 577–579, 1998.

[93] D. Heinke, Y. Sun, and G. W. Humphreys, "Modeling grouping through interactions between top-down and bottom-up processes: The grouping and selective attention for identification model (g-saim)," in *Attention and Performance in Computational Vision. Second International Workshop on Attention and Performance in Computer Vision (WAPCV 2004), LNCS 3368*, L. P. et al., Ed. Berlin, Heidelberg: Springer-Verlag, 2004, pp. 118–132.

[94] A. Treisman and S. Gormican, "Feature analysis in early vision: Evidence from search asymmetries," *Psychological Review*, vol. 95, no. 1, pp. 15–48, 1988.

[95] A. M. Treisman, "The perception of features and objects," in *Attention: Selection, Awareness, and Control*, A. Baddeley and L. Weiskrantz, Eds. Clarendon Press, Oxford, 1993, pp. 5–35.

[96] S. Engel, X. Zhang, and B. Wandell, "Color tuning in human visual cortex measured with functional magnetic resonance imaging," *Nature*, vol. 388, no. 6637, pp. 68–71, 1997.

[97] M. Begum, G. K. I. Mann, and R. G. Gosine, "A biologically inspired bayesian model of visual attention for humanoid robots," in *Proceedings of IEEE-RAS International Conference on Humanoid Robots*, 2006, pp. 587–592.

[98] M. Begum, F. Karray, G. K. I. Mann, and R. G. Gosine, "A probabilistic approach for attention-based multi-modal human-robot interaction," in *Proceedings of the*

18th IEEE International Symposium on Robot and Human Interactive Communication, 2009, pp. 200–205.

[99] ——, "A probabilistic model of overt visual attention for cognitive robots," *IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics*, vol. 40, no. 5, pp. 1305–1318, 2010.

[100] M. Z. Aziz, B. Mertsching, M. S. E.-N. Shafik, and R. Stemmer, "Evaluation of visual attention models for robots," in *Proceedings of the 4th IEEE Conference on Computer Vision Systems*, 2006, p. 20.

[101] T. Carron and P. Lambert, "Color edge detector using jointly hue, saturation, and intensity," in *Proceedings of IEEE International Conference on Image Processing*, 1994, pp. 977–981.

[102] U. Neisser, *Cognitive Psychology.* Prentice-Hall, Englewood Cliffs, NJ, 1967.

[103] F. Miau and L. Itti, "A neural model combining attentional orienting to object recognition," in *Proceedings of the 23rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2001, pp. 789–792.

[104] A. Salah, E. Alpaydin, and L. Akrun, "A selective attention based method for visual pattern recognition with application to handwritten digit recognition and face recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 3, pp. 420–425, 2002.

[105] D. Walther, U. Rutishauser, C. Koch, and P. Perona, "On the usefulness of attention for object recognition," in *Workshop on Attention and Performance in Computational Vision at ECCV*, 2004, pp. 96–103.

[106] N. Ouerhani and H. Hugli, "A model of dynamic visual attention for object tracking in natural image sequences," in *Proceedings of the Artificial and natural neural net-*

works 7th international conference on Computational methods in neural modeling, 2003, pp. 702–709.

[107] ——, "Robot self-localization using visual attention," in *Proceedings of IEEE International Symposium on Computational Intelligene in Robotics and Automation*, 2005, pp. 309–314.

[108] S. Frintrop, P. Jensfelt, and H. I. Christensen, "Attentional landmark selection for visual slam," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2006, pp. 2582–2587.

[109] A. Carbone, D. Ciacelli, A. Finzi, and F. Pirri, "Autonomous attentive exploration in search and rescue scenarios," in *WAPCV 2007*, L. Paletta and E. Rome, Eds. Springer-Verlag Berlin Heidelberg, 2007, pp. 431–446.

[110] C. Scheier and S. Egner, "Visual attention in a mobile robot," in *Proceedings of IEEE/RSJ International Symposium on Industrial Eelctronics*, 1997, pp. 48–52.

[111] S. Baluja and D. Pomerleau, "Using the representation in a neural network's hidden layer for task-specific focus of attention," in *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, 1995, pp. 133–139.

[112] C. Breazeal, A. Edsinger, P. Fitzpatrick, and B. Scassellati, "Active vision for sociable robots," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 31, no. 5, pp. 443–453, 2001.

[113] J. Weng, J. McClelland, A. Pentland, O. Sporns, I. Stockman, and E. Thelen, "Autonomous mental development by robots and animals," *Science*, vol. 291, no. 5504, pp. 599–600, 2001.

[114] J. Weng, "A theory for mentally developing robots," in *Proceedings of the 2nd International Conference on Development and Leanring (ICDL)*, 2002, pp. 131–140.

[115] J. Aloimonos, I. Weiss, and A. Bandyopadhyay, "Active vision," *International Journal of Computer Vision*, vol. 1, no. 4, pp. 333–356, 1988.

[116] D. H. Ballard, "Animate vision," *Artificial Intelligence*, vol. 48, no. 1, pp. 57–86, 1991.

[117] K. Brunnstrom, J.-O. Eklundh, and T. Uhlin, "Active fixation for scene exploration," *International Journal of Computer Vision*, vol. 17, no. 2, pp. 137–162, 1996.

[118] A. Maki, P. Nordlund, and J.-O. Eklundh, "Attentional scene segmentation: Integrating depth and motion," *Computer Vision and Image Understanding*, vol. 78, no. 3, pp. 351–373, 2000.

[119] H.-C. Nothdurft, "The role of features in preattentive vision: Comparison of orientation, motion and color cues," *Vision Research*, vol. 33, no. 14, pp. 1937–1958, 1993.

[120] C. Gilbert, M. Ito, M. Kapadia, and G. Westheimer, "Interactions between attention, context and learning in primary visual cortex," *Visual Research*, vol. 40, no. 10-12, pp. 1217–1226, 2000.

[121] P. J. Burt and E. H. Adelson, "The laplacian pyramid as a compact image code," *IEEE Transactions on Communications*, vol. 31, no. 4, pp. 532–540, 1983.

[122] J. G. Daugman, "Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters," *Journal of the Optical Society of America A. Optics and Image Science*, vol. 2, no. 7, pp. 1160–1169, 1985.

[123] B. Jahne, *Spatio-temporal image processing: Theory and scientific applications.* Springer-Verlag, 1993, pp. 81–84.

[124] E. H. Adelson and J. R. Bergen, "Spatiotemporal energy models for the perception of motion," *Journal of Optical Society of America. A. Optics and Image Science*, vol. 2, no. 2, pp. 284–299, 1985.

[125] D. J. Heeger, "Model for the extraction of image flow," *Journal of Optical Society of America. A. Optics and Image Science*, vol. 4, no. 8, pp. 1455–1471, 1987.

[126] E. P. Simoncelli and E. H. Adelson, "Computing optical flow distributions using spatio-temporal filters," M.I.T, Media Lab Vision and Modeling, Tech. Rep. 165, 1991.

[127] E. P. Simoncelli, "Distributed representation and analysis of visual motion," Ph.D. dissertation, Department of Electrical and Computer Science, Massachusetts Institute of Technology, 1993.

[128] M. Bravo and R. Blake, "Preattentive vision and perceptual groups," *Perception*, vol. 19, no. 4, pp. 515–522, 1990.

[129] S. Geman and D. Geman, "Stochastic relaxation, gibbs distributions, and the bayesian restoration of images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, no. 6, pp. 721–741, 1984.

[130] D. Geman, S. Geman, and P. Dong, "Boundary detection by constrained optimization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 7, pp. 609–628, 1990.

[131] Y. G. Leclerc, "Constructing simple stable descriptions for image partitioning," *International Journal of Computer Vision*, vol. 3, no. 1, pp. 73–102, 1989.

[132] T. Leung and J. Malik, "Contour continuity in region based image segmentation," in *Proceedings of the 5th Europe Conference on Computer Vision*, 1998, pp. 544–559.

[133] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.

[134] J. Malik, S. Belongie, T. Leung, and J. Shi, "Contour and texture analysis for image segmentation," *International Journal of Computer Vision*, vol. 43, no. 1, pp. 7–27, 2001.

[135] R. Adams and L. Bischof, "Seeded region growing," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 6, pp. 641–647, 1994.

[136] S. C. Zhu and A. L. Yuille, "Region competition: Unifying snakes, region growing, and bayes/mdl for multiband image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 9, pp. 884–900, 1996.

[137] J. R. Beveridge, J. Griffith, R. R. Kohler, A. R. Hanson, and E. M. Riseman, "Segmenting images using localizing histograms and region merging," *International Journal of Computer Vision*, vol. 2, no. 3, pp. 311–347, 1989.

[138] J.-M. Jolion and A. Rosenfeld, *A pyramid framework for early vision: Multiresolutional computer vision.* Kluwer Academic Publishers, 1994.

[139] P. Bertolino and A. Montanvert, "Multiresolution segmentation using the irregular pyramid," in *Flexibility and Constraint in Behavioral Systems*, R. J. Greenspan and C. P. Kyriacou, Eds. John Wiley and Sons, 1996, pp. 257–260.

[140] J. M. Jolion, "Stochastic pyramid revisited," *Pattern Recognition Letters*, vol. 24, no. 8, pp. 1035–1042, 2003.

[141] W. A. Phillips and W. Singer, "In search of common foundations for cortical computation," *Behavioral and Brain Sciences*, vol. 20, no. 4, pp. 657–722, 1997.

[142] P. Meer, "Stochastic image pyramids," *Computer Vision, Graphics, and Image Processing*, vol. 45, no. 3, pp. 269–294, 1989.

[143] P. C. Mahalanobis, "On the generalised distance in statistics," in *Proceedings of the National Institute of Sciences of India*, 1936, pp. 49–55.

[144] H. Jeffreys, "An invariant form for the prior probability in estimation problems," in *Proceedings of the Royal Society A. Mathematical and Physical Sciences*, 1946, pp. 453–461.

[145] S. Kullback and R. A. Leibler, "On information and sufficiency," *Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, 1951.

[146] A. Bhattacharyya, "On a measure of divergence between two statistical populations defined by their probability distributions," *Bulletin of the Calcutta Mathematical Society*, vol. 35, pp. 99–109, 1943.

[147] Z. Tu, X. Chen, A. L. Yuille, and S. C. Zhu, "Image parsing: Unifying segmentation, detection, and recognition," *International Journal of Computer Vision*, vol. 63, no. 2, pp. 113–140, 2005.

[148] A. Levin and Y. Weiss, "Learning to combine bottom-up and top-down segmentation," in *Proceedings of the European Conference on Computer Vision*, 2006, pp. 581–594.

[149] E. Borenstein and S. Ullman, "Combined top-down/bottom-up segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 12, pp. 2109–2125, 2008.

[150] A. Blake and M. Isard, *Active Contours: The Application of Techniques from Graphics, Vision, Control Theory and Statistics to Visual Tracking of Shapes in Motion.* Secaucus, NJ: Springer-Verlag New York, 1998.

[151] J. MacCormick, "Probabilistic modelling and stochastic algorithms for visual localisation and tracking," Ph.D. dissertation, Department of Engineering Science, University of Oxford, 2000.

[152] J. Duncan and G. W. Humphreys, "Visual search and stimulus similarity," *Psychological Review*, vol. 96, no. 3, pp. 433–458, 1989.

[153] R. P. N. Rao and D. H. Ballard, "An active vision architecture based on iconic representations," *Artificial Intelligence*, vol. 78, pp. 461–505, 1995.

[154] ——, "Object indexing using an iconic sparse distributed memory," in *Proceedings of the 5th International Conference on Computer Vision (ICCV)*, 1995, pp. 24–31.

[155] A. N. Redlich, "Redundancy reduction as a strategy for unsupervised learning," *Neural Computation*, vol. 5, no. 2, pp. 289–304, 1993.

[156] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape index," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 24, pp. 509–522, 2002.

[157] G. W. Humphreys, "Neural representation of objects in space: A dual coding account," *Philosophical Transactions of the Royal Society Lond. B. Biological Sciences*, vol. 353, no. 1373, pp. 1341–1351, 1998.

[158] M. R. Berthold and J. Diamond, "Constructive training of probabilistic neural networks," *Neurocomputing*, vol. 19, no. 1-3, pp. 167–183, 1998.

[159] B. Moghaddam and A. Pentland, "Probabilistic visual learning for object detection," in *Proceedings of the 5th International Conference on Computer Vision*, 1995, pp. 786–793.

[160] M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of Cognitive Neuroscience*, vol. 3, no. 1, pp. 71–86, 1991.

[161] Y. Amit and D. Geman, "A computational model for visual selection," *Neural Computation*, vol. 11, no. 7, pp. 1691–1715, 1999.

[162] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2001, pp. 511–518.

[163] L. Shams and J. Spoeslstra, "Learning gabor-based features for face detection," in *Proceedings of the World Congress on Neural Networks*, 1996, pp. 15–20.

[164] C. Papageorgiou and T. Poggio, "A trainable system for object detection," *International Journal of Computer Vision*, vol. 38, no. 1, pp. 15–33, 2000.

[165] C. Schmid and R. Mohr, "Local grayvalue invariants for image retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 5, pp. 530–535, 1997.

[166] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[167] Y. Ke and R. Sukthankar, "Pca-sift: A more distinctive representation for local image descriptors," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2004, pp. 506–513.

[168] A. Yuille, "Deformable templates for face recognition," *Journal of Cognitive Neuroscience*, vol. 3, no. 1, pp. 59–70, 1991.

[169] M. Weber, M. Welling, and P. Perona, "Unsupervised learning of models for recognition," in *Proceedings of the 6th European Conference on Computer Vision - Part I*, 2000, pp. 18–32.

[170] R. Fergus, P. Perona, and A. Zisserman, "Object class recognition by unsupervised scale-invariant learning," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2003, pp. 264–271.

[171] A. Mohan, C. Papageorgiou, and T. Poggio, "Example-based object detection in images by components," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 4, pp. 349–361, 2001.

[172] S. Agarwal, A. Awan, and D. Roth, "Learning to detect objects in images via a sparse, part-based representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 11, pp. 1475–1490, 2004.

[173] H. A. Rowley, S. Baluja, and T. Kanade, "Neural network-based face detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 1, pp. 23–38, 1998.

[174] H. Schneiderman and T. Kanade, "A statistical method for 3d object detection applied to faces and cars," in *Proceedings of IEEE Computer Soceity Conference on Computer Vision and Pattern Recognition (CVPR)*, 2000, pp. 746–751.

[175] V. Vilaplana, F. Marques, and P. Salembier, "Binary partition trees for object detection," *IEEE Transactions on Image Processing*, vol. 17, no. 11, pp. 2201–2216, 2008.

[176] H. Schneiderman, "Feature-centric evaluation for efficient cascaded object detection," in *Proceedings of IEEE Computer Soceity Conference on Computer Vision and Pattern Recognition (CVPR)*, 2004, pp. 29–36.

[177] S. Dickinson, H. Christensen, J. K. Tsotsos, and G. Olofsson, "Active object recognition integrating attention and viewpoint control," *Computer Vision and Image Understanding*, vol. 67, no. 3, pp. 239–260, 1997.

[178] T. S. Lee and M. Nguyen, "Dynamics of subjective contour formation in the early visual cortex," *Proceedings of the Natual Academy of Sciences of the United States of America*, vol. 98, no. 4, pp. 1907–1911, 2001.

[179] Y. Bar-Shalom and T. E. Fortmann, *Tracking and Data Association.* Academic Press, 1988.

[180] B. Bascle and R. Deriche, "Region tracking through image sequences," in *Proceedings of the 5th International Conference on Computer Vision*, 1995, pp. 302–307.

[181] M. Isard and A. Blake, "Condension - conditional density propagation for visual tracking," *International Journal of Computer Vision*, vol. 29, no. 1, pp. 5–28, 1998.

[182] A. Yilmaz, X. Li, and M. Shah, "Contour based object tracking with occlusion handling in video acquired using mobile cameras," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 11, pp. 1531–1536, 2004.

[183] C. F. Olson, "Maximum-likelihood template matching," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2000, pp. 52–57.

[184] Z. Jia, A. Balasuriya, and S. Challa, "Sensor fusion-based visual target tracking for autonomous vehicles with the out-of-sequence measurements solution," *Robotics and Autonomous Systems*, vol. 56, no. 2, pp. 157–176, 2007.

[185] C. R. Wren, A. Azarbayejani, T. Darrell, and A. P. Pentland, "Pfinder: Real-time tracking of the human body," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 780–785, 1997.

[186] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 5, pp. 564–577, 2003.

[187] S. Birchfield, "Elliptical head tracking using intensity gradients and color histograms," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1998, pp. 232–237.

[188] A. D. Jepson, D. J. Fleet, and T. F. El-Maraghi, "Robust online appearance models for visual tracking," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2001, pp. 415–422.

[189] Y. Lao, J. Zhu, and Y. F. Zheng, "Sequential particle generation for visual tracking," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 9, pp. 1365–1378, 2009.

[190] M. J. Black and A. D. Jepson, "Eigentracking: Robust matching and tracking of articulated objects using a view-based representation," *International Journal of Computer Vision*, vol. 26, no. 1, pp. 63–84, 1998.

[191] G. D. Hager and P. N. Belhumeur, "Efficient region tracking with parametric models of geometry and illumination," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 10, pp. 1025–1039, 1998.

[192] R. T. Colins, Y. Liu, and M. Leordeanu, "Online selection of discriminative tracking features," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1631–1643, 2005.

[193] Z. Yin and R. Collins, "Spatial divide and conquer with motion cues for tracking through clutter," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006, pp. 570–577.

[194] Y.-J. Yeh and C.-T. Hsu, "Online selecition of tracking features using adaboost," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 3, pp. 442–446, 2009.

[195] J. Wang, X. Chen, and W. Gao, "Online selecting discriminative tracking features using particle filter," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005, pp. 1037–1042.

[196] N. Bouaynaya and D. Schonfeld, "A complete system for head tracking using motion-based particle filter and randomly perturbed active contour," in *Proceedings of SPIE, Image and Video Communications and Processing*, 2005, pp. 864–873.

[197] Y. Li, H. Ai, T. Yamashita, S. Lao, and M. Kawade, "Tracking in low-frame-rate video: A cascade particle filter with discriminative observers of different lifespans," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007, pp. 1–8.

[198] C. Gu and M.-C. Lee, "Semiautomatic segmentation and tracking of semantic video objects," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 8, no. 5, pp. 572–584, 1998.

[199] R. Cutler and L. Davis, "Robust real-time periodic motion detection," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 22, pp. 781–796, 2000.

[200] S. Araki, T. Matsuoka, H. Takemura, and N. Yokoya, "Real-time tracking of multiple moving objects in moving camera image sequences using robust statistics," in *Proceedings of 14th International Conference on Pattern Recognition*, vol. 2, 1998, pp. 1433–1436.

[201] S. J. Julier and J. K. Uhlmann, "A new extension of the kalman filter to nonlinear systems," in *Proceedings of International Symposium Aerospace/Defense Sensing, Simulation and Controls*, 2001, pp. 182–193.

[202] G. Kitagawa, "Non-gaussian state-space modeling of nonstationary time series," *Journal of the American Statistical Association*, vol. 82, no. 400, pp. 1032–1041, 1987.

[203] C. Rasmussen and G. D. Hager, "Probabilistic data association methods for tracking complex visual objects," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 6, pp. 560–576, 2001.

[204] Y. Boykov and D. Huttenlocher, "Adaptive bayesian recognition in tracking rigid objects," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2000, pp. 697–704.

[205] Y. Yoon, A. Kosaka, and A. C. Kak, "A new kalman-filter-based framework for fast and accurate visual tracking of rigid objects," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 1238–1251, 2008.

[206] R. Rosales and S. Sclaroff, "3d trajectory recovery for tracking multiple objects and trajectory guided recognition of actions," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 1999, pp. 117–123.

[207] E. B. Koller-Meier and F. Ade, "Tracking multiple objects using the condensation algorithm," *Robotics and Autonomous Systems*, vol. 34, no. 2-3, pp. 93–105, 2001.

[208] A. Treptow, G. Cielniak, and T. Duckett, "Real-time people tracking for mobile robots using thermal vision," *Robotics and Autonomous Systems*, vol. 54, no. 9, pp. 729–739, 2006.

[209] T. Bando, T. Shibata, K. Doya, and S. Ishii, "Switching particle filters for efficient visual tracking," *Robotics and Autonomous Systems*, vol. 54, no. 10, pp. 873–884, 2006.

[210] S. Avidan, "Ensemble tracking," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005, pp. 494–501.

[211] Y. Liu and Y. F. Zheng, "Video object segmentation and tracking using ?-learning classification," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 8, no. 5, pp. 572–584, 1998.

[212] N. Ouerhani and H. Hugli, "A model of dynamic visual attention for object tracking in natural image sequences," in *International Work-Conference on Artificial and Natural Neural Networks (IWANN),LNCS*, vol. 2686, 2003, pp. 702–709.

[213] S. Frintrop and M. Kessel, "Most salient region tracking," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2009, pp. 1869–1874.

[214] G. R. Bradski, "Compter vision face tracking for use in a perceptual user interface," *Intel Technology Journal*, no. Q2, 1998.

[215] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, 2002.