FLORIDA STATE UNIVERSITY

COLLEGE OF ARTS AND SCIENCES

# CONTEXT-SENSITIVE SEMANTIC SEGMENTATION

By

NAN ZHAO

A Dissertation submitted to the
Department of Computer Science
in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Degree Awarded:
Spring Semester, 2015

Nan Zhao defended this dissertation on March 26, 2015.
The members of the supervisory committee were:

Xiuwen Liu

Professor Directing Dissertation

Ken Taylor

University Representative

Gary Tyson

Committee Member

Piyush Kumar

Committee Member

The Graduate School has verified and approved the above-named committee members, and certifies that the dissertation has been approved in accordance with university requirements.

To Those I Love and Those Who Love Me

# ACKNOWLEDGMENTS

I would like to express my sincere appreciation and special thanks to my adviser Dr. Xiuwen Liu for introducing me to this subject, encouraging my research, funding my trips to top conferences, and for helping me to grow as a research scientist. You have set an example of excellence as a researcher, mentor, instructor, and role model.

I would like to thank my committee members, Dr. Ken Taylor, Dr. Piyush Kumar, and Dr. Gary Tyson for serving on my committee. Thanks to you for your valuable comments and suggestions to my thesis work. I would also like to thank the Computer Science department for its persistent financial support that helps me finish my research.

I would also like to sincerely thank my undergraduate adviser Dr. Bingfa Li and my Master thesis adviser Dr. Jianzhou Zhang for their constant enthusiasm and encouragement. Without the priceless guidance and suggestions you have given to me, I wouldn't have been so lucky to insist on my research interests.

I would like to further thank my research collaborators Dr. Washington Mio, Dr. David Houle, Dr. Josef Allen, Mrs. Chaity Banerjee, Dr. Guiqing Hu, Mr. Zhongjun Hu, and Dr. Cheri Hampton. Thank you all for your research suggestions, your help with paper writing, data collection, data processing, and explanations of key concepts in your field to make them crystal clear to me. Without your support, I would not have had such many opportunities to build up and test my thesis in real applications.

I would like to thank my friends with similar research interests but with different academic background: Dr. Jingyong Su, Dr. Qian Xie and Dr. Zhengwu Zhang from the Department of Statistics for teaching me shape models, Markov Random Field, Conditional Random Field, expectation maximization, and belief propagation; Dr. Liangjing Ding from the Department of Scientific Computing for teaching me watershed segmentation and motion segmentation; Dr. Jonathon Bates, Dr. Qiuping Xu, Mrs. Mao Li, Dr. Wen Huang and Mr. Fangxi Gu from the Department of Mathematics for teaching me Laplacian, manifolds, differential equations, nonlinear dimension reduction, and variational segmentation; Dr. Luyang Wang from the Department of Physics for teaching me free energy theory; Dr. Aguang Dai from the Biophysics program for explaining to me cryo-electron

microscopy, missing wedge effects and back-projection; Dr. Frank Johnson from the Department of Psychology for teaching me human visual system and Gestalt grouping principles.

Special thanks go to my family. Words cannot express how grateful I am to my parents for all of the sacrifices that you have made on my behalf. Your prayer for me was what sustained me thus far. I would also thank my friends, Jiangbo Yuan, Yu Zhang, Nan Yang, Shuaiyuan Zhou, Jingyi Xiao, Yuanting Lu, Hongtao Yi, Guanyu Tian, Fangzhou Lin, Aguang Dai, Wei Zhang, Jia Ma, Xiaojun Zhu, Andy Pan, Weisu Wang, Di Shi, Jinbo Fan, Bo Sun, Xinhao Xu, Dan Wang among others, who have shared your happiness with me to make my life colorful in Tallahassee.

# TABLE OF CONTENTS

# LIST OF TABLES

# LIST OF FIGURES

# LIST OF ABBREVIATIONS

The following short list of abbreviations is used throughout the document, that I tried to use consistently.

| | |
|---|---|
| 2D | Two dimensional |
| 3D | Three dimensional |
| AMR | Average miss rate |
| ANN | Approximate nearest neighbor |
| BUS | Bottom-up segmentation |
| CBIR | Content based image retrieval |
| CRF | Conditional random field |
| DD | Data driven |
| DoG | Difference-of-Gaussian |
| GMM | Gaussian mixture model |
| HIV | Human immunodeficiency virus |
| IAFIS | Integrated automated fingerprint identification system |
| MAIF | Multiple assignment and inverted file |
| MCMC | Markov Chain Monte Carlo |
| MRF | Markov random field |
| NN | Nearest neighbor |
| NMS | Non-maximum suppression |
| PDE | Partial differential equation |
| SNR | Signal to noise ratio |
| TDS | Top-down segmentation |
| TPS | Thin plate spline |

# ABSTRACT

Recognizing and representing objects of certain categories become increasingly important due to the availability of high-resolution imaging technologies and the explosive amount of digital data. In particular, semantic segmentation of given data (i.e.: two dimensional images or three dimensional volumes) labels or extracts objects in the form of contiguous regions with similar semantic interpretation. Hence semantic segmentation offers great rewards from object recognition and image segmentation. However, the combination of difficulties from both fields also yields incredible computational challenges. In practice, the appearance of objects is under the influence of views, poses, colors, shapes, scales, occlusion, illumination conditions and intrinsic imaging limitations. Thus an ideal semantic segmentation should tolerate both the considerable intra-class variance and the noticeable inter-class similarities in terms of appearance.

The primary contribution of this thesis is the investigation on context cues that may improve semantic segmentation. I first propose a novel two-stage framework to solve a special problem of semantic segmentation, in which the target objects are much more likely to be observable under the existence of context objects. In the first stage, global salient context objects are segmented using appearance features. The second stage formulates multiple types of context cues, followed by a model that combines both appearance and context cues. I then apply this framework to the problem of spike segmentation and tattoo segmentation, resulting in a cryo-electron tomogram segmentation system and a tattoo classification system. The first system allows biophysicists to significantly accelerate their data processing by replacing manual annotation with semi-automatic segmentation, whereas the second system explores for the first time the possibility of category-level tattoo classification by machine. As shown by these two systems, the proposed models outperform traditional object-centered models that purely focus on appearance features.

# CHAPTER 1

# INTRODUCTION

## 1.1 Motivation

Image segmentation is one of the most fundamental and challenging problems in computer vision. It is aimed at simplifying and/or changing the representation of a two-dimensional (2D) or three dimensional (3D) image into some form that is more efficient for further processing such as recognition, retrieval, and reconstruction. It has been demonstrated that segmentation and recognition mutually benefit each other when they are combined in a single task, namely **semantic segmentation** [118, 3]. More precisely, each target object is represented as a contiguous region with similar semantic interpretation in semantic segmentation. The association between each pixel/voxel and the object is represented by one of the $K$ labels, where $K$ is the number of possible objects in the given data. Take images in Fig. 1.1 as an example. The first row shows four exemplar 2D images of cars. If the cars in the images are the target objects, then the ground, the sky and the buildings are the background. So the corresponding results of semantic segmentation on cars are shown in the second row, in which the background regions are all marked as black.

Mathematically, semantic segmentation intends to find a general 'function' $f$ such that

$$y = f(x), \tag{1.1}$$

where $x$ is the original data (image) and $y$ is the segmentation result (label map). In order to achieve semantic segmentation, there are two key components in this problem: localization and extraction of the interested objects. Object localization is a recognition process that finds out where the objects are and object extraction is a segmentation process that separates the objects from the background. Therefore, the function $f$ can be further decomposed into the following form in terms of three general functions:

$$f(x) = g^n(l, d)(x). \tag{1.2}$$

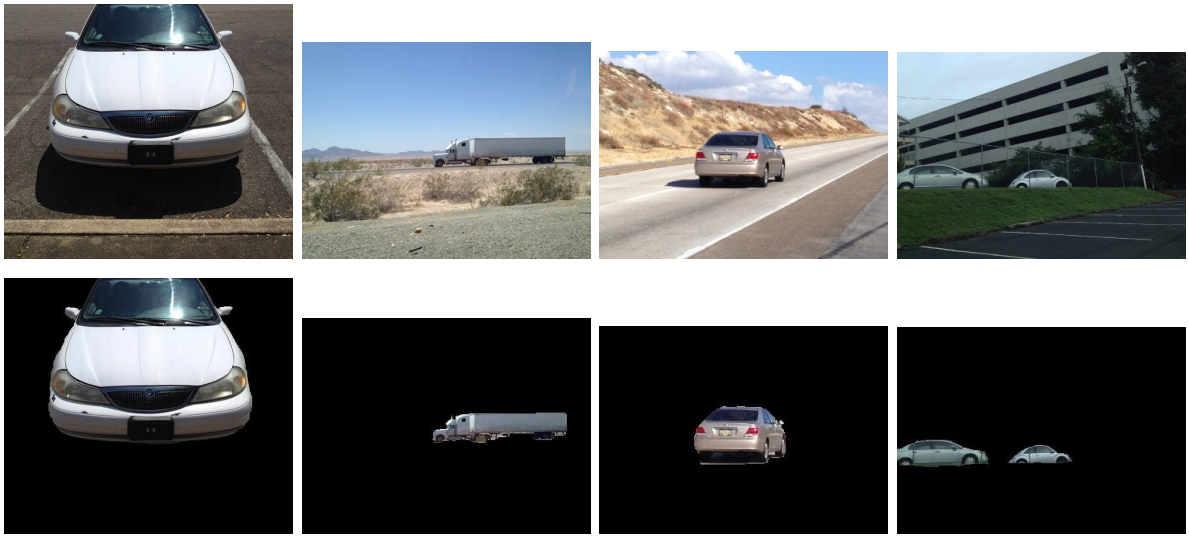


Figure 1.1: Four semantic segmentation examples. Semantic segmentation on an image groups together the pixels in a contiguous region with common semantic meaning. The first row shows four 2D images, with their respective semantic segmentation on cars in the second row.

Here the recognition function $l(.)$ extracts the target object(s) and the segmentation function $d(.)$ localizes the target object(s); function $g(f_1, f_2)(x)$ denotes any relationship between two functions, such as $f_1(f_2(x))$ and $f_2(f_1(x))$; $g^n$ is defined as the $n'th$ iteration of $g$, where $n$ is a positive integer,

$$g^1 = g \tag{1.3}$$

and

$$g^{n+1} = g \cdot g^n. \tag{1.4}$$

In the past few decades, semantic segmentation has managed to tackle a wide range of real-world compute vision problems. For instance, it has been successfully applied in face recognition [127, 50], pedestrian detection [74], iris recognition [54], and tumor detection [100]. It has also provided powerful tools to address many challenging problems, such as video surveillance, scene analysis, traffic control, autonomous car driving, and content-based image retrieval.

## 1.2 Challenges

There are two issues that need to be tackled to make semantic segmentation more successful – the computational complexity and the data quality. Traditional semantic segmentation techniques [22, 30, 40, 121, 130, 41, 89] are primarily based on appearance features, such as intensity, color, shape, texture and edge response. While the techniques of [22, 40, 121, 41] work well for small 2D/3D dataset, these globalization-based techniques become computationally intractable and inapplicable when faced with a large scale, such as an image with one million pixels. For example, recent advances in a few imaging technologies with high resolution stimulate a great interest from the biology community in modeling and analyzing biological structures that are too small to be observed in the past. However, it is quite expensive to manually extract these structures for further processing because significant human effort is required. Semantic segmentation on a large scale thus becomes a bottleneck of the research on these biological structures. In addition, the performance of semantic segmentation highly depends on the quality of the input data. With an extremely low signal to noise ratio (SNR), most of the state-of-the-art methods in the literature, especially those without globalization [30, 130, 89], fail to accurately localize small objects in the data. This is primarily because of the appearance similarities between the small objects and the background noise. Because globalization is computationally expensive and appearance features are noise-sensitive, segmentation on small objects with low SNR is often achieved by human labor.

## 1.3 Related Work

This section reviews the methods that are related to semantic segmentation and attempt to overcome the challenges mentioned in the previous section. The problems of each method are summarized at the end of its introduction. In general, the related methods can be clustered into three approaches: bottom-up segmentation (BUS), top-down segmentation (TDS) and the combination of BUS and TDS.

The dominant and earliest segmentation approach is carried out in a bottom-up manner. It often starts without the help from any model of the target object. Specifically, pixels in the image are grouped into a number of homogeneous regions in terms of image-dependent and object-independent local features such as texture, edge and color. Some globalization criteria often cooperate with the local features to produce the segmentation as close to the object contours as possible. In order

to achieve semantic segmentation, a recognition step is then applied to label regions as different objects according to their image-independent and object-dependent features. In the most basic form of bottom-up approach (when $n = 1$), we have $y = f(x) = l(d(x))$, which means recognition is under the facilitation of segmentation.

Another approach is carried out in a top-down manner, primarily guided by the engineered or learned models of the target object. Given an image, the target objects are localized at first and then extracted under the guidance of the appearance prior, such as shape and texture. In its basic top-down manner (when $n = 1$), we have $y = f(x) = d(l(x))$, which means segmentation is under the facilitation of recognition.

As each region of interest is usually associated with a semantically meaningful object, semantic segmentation is more challenging than the BUS. Even though the computation of the low-level local features is often efficient, the BUS only favors the low-level homogeneity in each region and thus often produces over-segmentation or under-segmentation results due to the lack of object class models. In contrast, the process of semantic segmentation requires both bottom-up and top-down cues in order to satisfy the homogeneity in terms of not only low-level texture but also high-level semantic meaning. Therefore, object classes should be modeled appropriately for semantic segmentation. However, due to the potential large intra-class variance in terms of object shape and appearance, it is often very difficult to generate a group of top-down cues that describe the object classes very well. Conversely, the shape and appearance of object parts from difference object classes may be similar. Hence it is often very difficult to obtain bottom-up cues that are able to distinguish similar regions from different classes.

Regarding the problems of BUS and TDS, it is difficult to obtain a model for semantic segmentation using bottom-up or top-down approach alone. Therefore, a well-defined function $y = g(l, d)(x)$ is necessary for semantic segmentation. There has been some sparse work in this direction, such as OBJCUT ([67]), image parser ([126]) and jigsaw ([19]). In order to give a quick review on the problems and the potentials of the state of the art methods in solving semantic segmentation, we summarize the basic bottom-up methods (mainly focusing on the design of $d(x)$), the primary top-down methods (concentrating on $l(x)$) as well as the methods combining these two (formulating $g(f_1, f_2)(x)$) in the following sections.

### 1.3.1 Bottom-Up Segmentation

The most straightforward way of segmentation is to find a threshold for splitting an image into regions in the form of connected components. However, due to the variance in illumination and intra-class statistics, a single threshold is rarely enough for image segmentation.

**Watershed.** An efficient alternative on gray scale images is watershed computation. The concept of watershed comes from topography. A watershed line is a ridge of land that divides two adjacent river systems (normally called catchment basins). In the watershed algorithm, an image is interpreted as a topographic surface where the gray level for each pixel represents its altitude. The goal of this algorithm is to segment an image into several catchment basins, which are homogeneous in the sense that from all pixels inside the same catchment basin we can go downhill to find the basin's bottom (with minimum altitude). Therefore, the catchment basins correspond to the regions of the segmented image and the high watersheds correspond to the region boundaries. An efficient approach to watershed segmentation algorithm is to start flooding the catchment basins from the bottom (all of the local minima). Through a breadth-first search, pixels belonging to potential catchment basin members are put into a priority queue for further labeling in each flooding level. Finally, ridges are labeled wherever two evolving regions meet ([130]). Since watershed computation relies on the difference in altitudes between regions and ridges, it is usually applied on the smoothed gradient magnitude images and color images. However, watershed segmentation suffers from over-segmentation in that each local minimum is associated with a unique region. Thus it is usually used in an interactive segmentation system, where the local minima are replaced by user's markers.

**Graph-based Merging.** Felzenszwalb and Huttenlocher [40] proposed a segmentation algorithm based on Kruskal's minimum spanning tree (MST), which consists of a number of edges selected from a graph. In Kruskal's algorithm, all edges are unmarked at first. To generate a Kruskal's MST, the unmarked edge with the minimum weight keeps on being marked if it does not close a circuit of all the marked edges until every node of the graph is reached out. In [40], each pixel is a node in the graph and weights of each edge measure the dissimilarity between pixels. Since an MST is a subset of a graph that connects all the nodes by edges with minimum-weights and without any cycle, this algorithm is also called a graph-based approach. Edges in each tree are taken into account in an increasing order of weight; any two adjacent regions are merged into a region if the graph maintains cycle-free, and if their difference is smaller than their minimum internal

difference. By merging regions relying on the decreasing order of edge weights, the segmentation result is thus neither too coarse nor too fine. However, it is difficult to involve multiple features in this model to assist semantic segmentation. Thus it is limited to segmentation where one feature is distinctive enough for semantic segmentation.

**Normalized Cuts.** This algorithm models pixels as vertices in a weighted undirected complete graph $G = (V, W)$, where the weight $w_{ij}$ of an edge $(i, j) \in W$ represents the similarity between vertices $i$ and $j$. Based on this model, the problem of segmentation can be formulated as a graph partitioning problem, trying to find a partition $V_1, V_2, ..., V_k$ of the vertex set $V$ such that, according to some measure, the vertices in any partition $V_i$ are highly similar whereas any pair of vertices from two different partitions have low similarity. Considering figure-ground semantic segmentation, the aim is to partition a graph $G = (V, W)$ into two disjoint sets, the figure $A$ and the background $B$, by removing a group of edges connecting these two sets. This group of eliminated edges is called a *cut* in graph theory, and the total weight of these edges reflects the degree of similarity between $A$ and $B$:

$$cut(A, B) = \sum_{i \in A, j \in B} w_{ij}. \tag{1.5}$$

Consequently, the figure-ground segmentation problem becomes finding the *minimum cut* among a set of potential *cuts*. To overcome the bias toward a cut to the edges between a small set of isolated vertices and the remaining ones, the measure of cut cost is replaced by so-called *normalized cut (Ncut)*, a fraction of the cut $cut(A, B)$ to all the vertices $V$ in the graph ([121]):

$$Ncut(A, B) = \frac{cut(A, B)}{assoc(A, V)} + \frac{cut(A, B)}{assoc(B, V)}, \tag{1.6}$$

where $assoc(A, V) = \sum_{i,t} w_{it}$ is the total weight of edges from vertices in A to all vertices in the graph G and $assoc(B, V)$ is similarly noted. Through this disassociation between two sets, even the cut partitioning out small isolated vertices will have large $Ncut$ value in that the cut value will be the major contribution to the connections from that small set to the remaining vertices. For $cut(A, B)$ in the graph $G$, $Ncut(A, B)$ measures the similarity between the figure and ground. Similarly, the total similarity of vertices in the same set can be measured by normalized association:

$$Nassoc(A,B) = \frac{assoc(A,A)}{assoc(A,V)} + \frac{assoc(B,B)}{assoc(B,V)}. \tag{1.7}$$

Since $Ncut(A,B) = 2 - Nassoc(A,B)$, a cut that minimizes the similarity between figure and ground also maximizes the total similarity of vertices belonging to the same set simultaneously. Let $x$ be an indicator vector where $x_i = +1$ if node is in A and $x_i = -1$, otherwise. Let $d(i) = \sum_j w_{ij}$, $D$ be an $N \times N$ diagonal matrix with $d$ on its diagonal, and $W$ be an $N \times N$ symmetric matrix with $W_{ij} = w_{ij}$, it was proved that

$$\min_{(A,B)} Ncut(A,B) = \min_{y} \frac{y^T(D-W)y}{y^T D y}, \tag{1.8}$$

subject to the constraints that $y = ((1+x) - b(1-x))/2$ and $y^T D1 = 0$. Minimizing $Ncut(A,B)$ subject to the constraint above is an NP-hard problem. However, if $y$ is relaxed to take real values, the optimal normalized cut solution can be approximated by bi-partitioning the graph with the eigenvector $y$ corresponding to the second smallest eigenvalue of a generalized eigenvalue system:

$$(D-W)y = \lambda D y. \tag{1.9}$$

Specifically, a splitting point can be chosen for bi-partition in multiple ways such as a constant value (0 or 0.5), the median value or the value that minimizes $Ncut(A,B)$. After the graph is partitioned into two pieces, subdivision can be applied to each piece and repartition could repeat until the value of Ncut is larger than a given threshold, indicating the non-existence of clear splitting point, or the number of vertices in the piece is smaller than a given threshold. Even though normalized cut can extract the salient contours regardless of the clustered background, it is impractical to be applied on data with high resolution. As matrix $D$ becomes too large because of the resolution, solving such an eigenvalue system would be too expensive.

**K-means.** The methods introduced so far are deterministic, which means semantic segmentation is formulated as a deterministic optimization problem, iteratively moving edges between foreground and background toward an optimal location. Thus they share the problem of the deterministic approach – it lacks a general formulation that naturally allows arbitrary number of distinctive features. To overcome this limit, semantic segmentation is also regarded as a stochastic optimization problem: the probability distribution of the label variable is repeatedly estimated for

each pixel. The most well-known statistic algorithm is K-means – a classical clustering technique that clusters a data set into $K$ clusters, where $K$ is given as a prior. When it is applied to segmentation, an image is modelled as a parametric model of a probability density function, which is the mixture of several underlying spherical symmetrical probability distributions in terms of the Euclidean distances from their centers to pixels [86]. Each spherical symmetrical distribution corresponds to a cluster and feature vectors in each cluster are thus samples from the corresponding distribution. K-means segmentation is aimed at breaking the image into regions while attempting to minimize square error, the sum of square Euclidean distances in feature space between pixels labelled as a cluster and the center of that cluster. Initialized from randomly chosen $K$ pixels from the input feature space as $K$ cluster centers, it then iteratively assigns each pixel to the nearest cluster followed by updating the location for each cluster center as the centroid of each cluster until convergence. However, the segmentation performance of K-means depends on the choice of initial cluster centers. These centers are hence usually initialized by selecting random seeds with at least center distance $D_{min}$ between each other or using more sophisticated methods such as random partition ([51]) and K-means++ ([5]). Another issue of the initialization is that the number of component distributions $k$ could be unknown in real-world applications ([41]). In such case, $k$ should be estimated from the data through strategies such as trying a number of $k$ or minimizing a coding length ([89]). Last but not the least, K-means segmentation is also sensitive to outliers because of its minimization on a sum of squared Euclidean distances of pixels from their corresponding cluster centers. K-medoids [64] was proposed to make it less sensitive.

**Gaussian mixture model.** Gaussian mixture model (GMM) is another machine learning algorithm that has been used as a tool for statistical segmentation. In this model, pixels in the given image are assumed to be samples from an underlying parametric model, the superposition of several Gaussian density functions:

$$p(x|\alpha_k, \mu_k, \Sigma_k) = \sum_k \alpha_k \mathcal{N}(x|\mu_k, \Sigma_k), \tag{1.10}$$

where $\alpha_k$, $\mu_k$ and $\Sigma_k$ are the mixing coefficient, mean and covariance for the $k$th Gaussian density function respectively:

$$\mathcal{N}(x|\mu_k, \Sigma_k) = \frac{e^{-d^2(x,\mu_k,\Sigma_k)}}{|\Sigma_k|}. \tag{1.11}$$

Similar to K-means, each Gaussian distribution corresponds to a segmented region (cluster) and pixels in each region are thus samples from the corresponding distribution; in contrast, GMM-based segmentation is aimed at splitting the image into k regions while finding the maximum likelihood estimate of a mixture of Gaussian distributions. Instead of using the Euclidean distance in the feature space, Mahalanobis distance

$$d(x_i, \mu_k, \Sigma_k) = \|x_i - \mu_k\|_{\Sigma_k^{-1}} = \sqrt{(x_i - \mu_k)^T \Sigma_k^{-1} (x_i - \mu_k)} \tag{1.12}$$

is often used in Gaussian mixture models, where $x_i$ is the feature vector for the $i$th pixel, $\mu_k$ is the center for the $k$th cluster, and $\Sigma_k$ is the covariance estimate for the $k$th cluster. GMM-based segmentation then entails approximating the GMM and labelling each pixel to the region with the highest likelihood. Model estimation and segmentation can be coupled together by the expectation maximization (EM) algorithm ( [35]), which is a greedy descent algorithm that iterates between them to carry out maximum likelihood estimation (MLE) of GMM. There are two steps in EM: the expectation step estimates the likelihood of each sample generated by each Gaussian distribution; the maximization step updates the mixing coefficient, mean and covariance for each Gaussian distribution. However, GMM suffers from the scale difference between the target object and the background object. If the background object is much larger than the target object, it is more than likely that the estimated GMM only describes the background object. Thus this approach is not applicable for small object segmentation in cluttered background. Again, this model also suffers from the initialization issue that the number of component distributions $k$ may be difficult to estimate in real-world applications.

**Mean shift.** Followed by the idea that a complicated probability density function for an image can be decomposed into a group of Gaussian probability distributions, it is natural to consider if pixels could be labelled relying on some properties of GMM without explicitly representing the probability density function. A way that is straightforward for such purpose is to seek major peaks in the image data distribution, rather than finding a parametric GMM. The main idea of this smooth non-parametric model is to label pixels climbing up to the same peak as in the same region. One of the most widely used approaches for estimating the density function is called Kernel density estimation or Parzen windows [36]. In this approach, we convolve the data with a given kernel

$$f(x) = \sum_{x_i \in N(x)} K(x_i - x), \tag{1.13}$$

where $x_i$ is the $i$th sample, $N(x)$ is the neighborhood of $x$ defined as a Parzen window (outside of which $K(x) = 0$), and $K(t)$ is a kernel function, e.g.: a Gaussian kernel, which is

$$K(x_i - x) = e^{c\|x_i - x\|^2}. \tag{1.14}$$

Optimization methods like gradient descent can be applied later to find local maxima. However, in the case of high-dimensional search space or even low-dimensional but extremely sparse space, it is very difficult to evaluate the density function $f(x)$. To overcome this problem, another optimization technique named multiple restart gradient descent is used instead, where an input feature vector $x_i$ can be randomly picked up from samples as an initial local maximum $p_0$. Mean shift then computes the gradient of the kernel density function $f(x)$ centered at $p_0$ followed by climbing up the hill in that direction. Specifically, the gradient of the kernel density function is

$$\nabla f(x) = \sum_{x_i \in N(x)} (x_i - x) G(x - x_i) = \sum_{x_i \in N(x)} (x_i - x) g\left(\frac{\|x_i - x\|^2}{h^2}\right), \tag{1.15}$$

where $g(t) = -k'(t)$ and $k'(t)$ is the first derivative of $k(t)$. Therefore, the weighted mean of the current neighborhood ($N(x)$) can be written as

$$m(x) = \frac{\sum_{x_i \in N(x)} x_i G(x_i - x)}{\sum_{x_i \in N(x)} G(x_i - x)}. \tag{1.16}$$

The gradient expression can then be re-written as:

$$\nabla f(x) = \left[\sum_{x_i \in N(x)} G(x_i - x)\right] d(x), \tag{1.17}$$

where $d(x) = m(x) - x$ is so called the mean shift. In iteration $k$ of mean shift, $p_k$ is replaced by $m(p_k)$, recorded as $p_{k+1}$ for the next iteration. This process repeats until it converges to a local minimum of the data distribution $f(x)$. However, as proved by [30], regular gradient descent cannot guarantee the convergence unless proper step size is chosen.

**Graphical model.** The statistic approach works under the existence of distinctive features. Unfortunately, it may be difficult to engineer ideal features that make our target object distinctive. Thus interactive segmentation is a strong tool to assist object localization and object feature learning. The user inputs are usually modeled as additional information combined appearance features in a graphical model. A common objective of image segmentation is the desire to group pixels with similar appearance while having boundaries between regions of short length and across visual discontinuity. If we restrict the boundary measurements to be direct neighbors and compute region membership statistics via summing over pixels in regions manually chosen as either the foreground or the background, we can formulate the segmentation as an energy function using either regularization or binary Markov random field (MRF). An early example of a discrete labelling problem that combines both boundary-based and region-based energy term was proposed by [70], deriving the energy function from minimum description length (MDL) coding. Given $\delta(f_1(x) - f_2(x))$ is 0 if $f_1(x) = f_2(x)$ and 1 otherwise, the segmentation problem can be modelled as minimizing a combination of a region term and a boundary term:

$$\begin{aligned} E(f) &= \sum_{i,j} E_r(i,j) + E_b(i,j), \text{ where} \\ E_r(i,j) &= E_S(I(i,j); R(f(i,j))), \text{ and} \\ E_b(i,j) &= s_x(i,j)\delta(f(i,j) - f(i+1,j)) + s_y(i,j)\delta(f(i,j) - f(i,j+1)). \end{aligned} \tag{1.18}$$

The region term $E_r(i,j)$ measures the coherence between the intensity value (or color) $I(i,j)$ of pixel and the statistics of region $R(f(i,j))$ chosen by user. Here $R(f(i,j))$ can simply be the mean in gray level or color domain or be more complicated, such as region-based intensity value histograms [22] or GMMs in color space [114]. For the boundary term $E_b(i,j)$, it measures the agreement between neighboring pixels proportioned by $s_x(i,j)$ and $s_y(i,j)$, which are horizontal and vertical smoothness terms respectively. Normally the strength of the smoothness term is inversely proportional to the discontinuity between neighboring pixels [114].

Generally, the gradient decent technique could be applied iteratively to minimizing the energy function above. However, drawbacks of this technique are that it is slow as well as potentially reaching local minima. Even though there are several known techniques for MRF-based energy minimization, graph cuts is the most widely used one. Boycov and Jolly [22] were the first to apply

this technique based on binary Markov random field (MRF) for figure-ground segmentation problem. In their method, seeds (pixels) from foreground and background are sampled by a user via an image brush. The statistic priors (intensity or color histogram) can then be learned from these foreground and background seeds. The nodes that are more compatible with either foreground or background seeds will have a stronger link to the corresponding terminal. Meanwhile, neighboring pixels with greater smoothness get stronger connections. In this way, image segmentation is modelled as a minimum-cut/maximum-flow problem which can be solved in polynomial time and is usually called min-cut for short. Each node will finally be assigned as either foreground or background relying on the terminal to which they remain linked.

One major extension to the original figure-ground segmentation approach of [22] is GrabCut segmentation system ([114]), where the region statistics are modelled as a GMM and user input is minimized through a bounding box. The pixels around and inside the box outline are regarded as the background and the foreground seeds respectively. The system iteratively re-estimates the statistics of the interior region so that the foreground color model will migrate toward a better estimate. Additional manual refinement is also allowed afterwards. However, the computational cost of the graphical model increases dramatically with the resolution of the data [7].

### 1.3.2 Top-down Segmentation

Even though the state-of-the-art segmentation algorithms using bottom-up cues provide impressive results, their difficulties in semantic segmentation are still obvious. Despite the fact that BU segmentation can be generally applied to any image in order to find image discontinuities that indicate potential object boundaries, their major problems, however, include splitting an object into regions and merging objects with the background. These drawbacks inherit from inevitable ambiguities that can hardly be distinguished without prior knowledge of the object class, due to the large intra-class variance in terms of color, texture, etc. for most objects. In addition, local parts of a salient object are not necessarily salient in contrast to their background. The less salient object parts may be merged with background.

Concerning these problems, another trend of figure-ground semantic segmentation is that of a top-down visual process, in which segmentation is primarily based on the guidance of object representations generated in the high-level: an object recognition step, sometimes called salient object detection ([83]), is applied at first to identify which specific class the detected object belongs

to. Then the object is segmented from its background aided by prior knowledge of the object class in terms of its possible shape (contour) and appearance (patch). In other words, according to TDS approach, segmentation is under the facilitation of recognition.

**Parametric active contour.** One way of TDS is to explicitly model the target object as a parametric contour, for which a contour is initialized in the given image and evolves toward the solution under the guidance of image discontinuities, such that both internal forces such as smoothness constraints and external forces such as high level shape constraints are taken into account. Besides cues in color space and spatial space, contours of objects can also be considered as features for segmentation. In contour based segmentation, a segmentation of the image plane $\Omega$ is achieved by locally minimizing the energy (or cost) of a curve relying on how well it fits the desired contour. Invented by Kass et al. [63], a parametric active contour, named Snake, is defined as an energy minimizing spline that evolves towards the closed contour of an object in the image. The initial shape and location for such snake should be given near the desired contour via some priors like human input, high level interpretations or data adjacent in spatial or time domain. Let the position of a snake be represented explicitly as a parametric form $f(s) = (x(s), y(s))$, where $x(s), y(s)$ are x, y coordinates along the spline and the arc-length $s \in [0, 1]$. Then the energy function of a snake is

$$\begin{aligned} E^*_{snake} &= \int_0^1 E_{snake}(f(s))\, ds \\ &= \int_0^1 \Big( E_{internal}(f(s)) + E_{image}(f(s)) + E_{constraint}(f(s)) \Big)\, ds, \end{aligned} \tag{1.19}$$

where

$$\begin{aligned} E_{internal}(f(s)) &= \alpha(s) \left| \frac{df}{ds} \right|^2 + \beta(s) \left| \frac{d^2 f}{ds^2} \right|^2, \\ E_{image}(f(s)) &= w_1 E_{line} + w_2 E_{edge} + w_3 E_{termination}. \end{aligned} \tag{1.20}$$

The first term is the internal spline energy providing a smoothness constraint on the snake. There are mainly two causes related to the change of smoothness: stretching and bending. Thus it is composed of a first-order term controlled by a measure of the elasticity or the tension along the snake, $\alpha(s)$, and a second-order term controlled by a measure of the rigidity of the snake, $\beta(s)$.

Therefore, a large $\alpha(s)$ penalizes distance variance between contour points, whereas a large $\beta(s)$ penalizes oscillations in the contour.

The second term is the external energy deriving from the image where the snake lies. It is a weighted combination of three terms in that snake may be attracted to lines, edges and terminations in an image. The line term is commonly defined as $E_{line} = I(x, y)$ where $I(x, y)$ denotes the image gray level at location $(x, y)$. Therefore, if $w_1$ is positive the snake is attracted to light lines and if negative then it is attracted to dark lines. The edge term is defined by $|\nabla I(x, y)|^2$, which attracts the snake towards edges with large image gradients. The termination term pushes the spline toward line terminations and corners. Let $G(x, y) = S_\alpha(x, y) * I(x, y)$ be a smoothed version of the image $I$ and $\theta = \arctan(G_y/G_x)$ be the gradient angle, unit vectors along and perpendicular to the gradient direction can then be represented by $n = (\cos\theta, \sin\theta)$ and $n' = (-\sin\theta, \cos\theta)$ respectively. Following this, the termination term can be written as

$$\begin{aligned} E_{termination} &= \frac{\partial\theta}{\partial n'} \\ &= \frac{\partial^2 G/\partial n'^2}{\partial G/\partial n} \\ &= \frac{(\partial^2 G/\partial y^2)(\partial G/\partial x)^2 - 2(\partial^2 G/\partial x\partial y)(\partial G/\partial x)(\partial G/\partial y) + (\partial^2 G/\partial x^2)(\partial G/\partial y)^2}{((\partial G/\partial x)^2 + (\partial G/\partial y)^2)^{3/2}}. \end{aligned} \tag{1.21}$$

The third term is responsible for imposing external constraints such as springs attached by the user or high level shape information. On one hand, if the snake is near the desired local minimum, the constraint term will pull the snake even closer. On the other hand, if the local minimum where the snake locates is regarded as incorrect via a high level process, the constraint term will force the snake away to another local minimum nearby. For example, a spring-based constraint term can be defined as $E_{constraint} = k_i\|f(i) - p(i)\|^2$ where $p(i)$ is the $i'th$ anchor point.

Parametric active contour overcomes both the computational inefficiency and low SNR due to its concerning on only the adjacent but global region of the active contour. However, it suffers from the fact that it requires a sophisticated re-griding process to eliminate overlap of control or marker points. Moreover, it lacks a meaningful statistical framework for extensional use [31].

**Geometric active contour.** Another way of TDS is to model the target object as a geometric contour. Instead of being explicitly represented as a curve function, the contour is represented in an

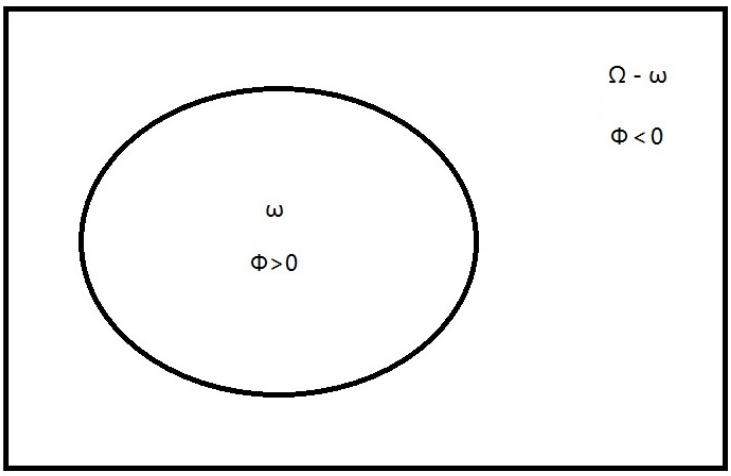



Figure 1.2: A curve, defined by the zero level set of the function $\Phi(\cdot)$, is the boundary between regions where $\omega : \Phi(x, y) > 0$ and $\Omega - \omega : \Phi(x, y) < 0$.

implicit manner. In level set segmentation ([92], [128], [37]), curves are defined as the zero crossing (also called zero level set) of some higher dimensional embedding function $\Phi$ in the image domain $\Omega$. An example level set function is shown in Fig. 1.2. The main idea of level set segmentation is to update this time-dependent embedding function $\Phi(x, t)$ rather than the curve function $f(s)$ in each step. Consequently, the zero level set of each embedding function ($\Phi(f(t), t) = 0$) defines the curve that is propagated toward contours of the desired objects in the current step. The main advantage of level set is that the zero level set may change topology (break, merge, etc.) during the evolution while the embedding function always remains as a function. Cremers et al. [31] present a wonderful survey of level set segmentation. However, it suffers from the regularization problem that shapes the level set function properly. It is tricky to decide when to re-shape the level set function and thus produce desired contours [92].

**Interactive contour extraction.** Active contour methods allow a user to give a rough boundary of the interest and then have such a boundary evolve toward the desired contour. Some user input constraints are often required for a desired result. As an alternative, intelligent scissor developed by [99] allows real-time interaction, the contour keeps on being optimized while the user is drawing. As shown in Fig. 1.3, when the user inputs a rough boundary, a better curve clinging to the desired contour (green curves) is generated. To compute the optimized contour of the interest,

each point in the image is associated with a cost indicating its potential of being a contour point. Such cost is based on an $N_8$ neighborhood and is computed by concerning not only the zero-crossing but also the gradient in terms of its magnitude and orientation. When a user keeps on drawing, the system continues to compute the path with the lowest cost from the starting seed point to the current point. To prevent the curve from jumping around arbitrarily, the curve will be "frozen" after it is stabilized for a period and the ending point of such curve is assigned as the new staring point. Moreover, the optimal curve may jump onto the boundary with high contrast nearby. Thus the intensity profile of the current optimized curve is learned as a constraint encouraging the curve moving along current boundary. However, it is difficult to be extended into 3D due to the amount of human workload involved.

**Semi-global deformable template matching.** In contrast to use initial contours to avoid target object localization, semi-global deformable templates are usually for localization, in the form of local patches with certain configuration. A collection of local patches for object class, a codebook, is a visual dictionary with a large number of visual words. Each class of objects can be described by a set of visual words. Leibe and Schiele [75] propose a probabilistic formulation for segmenting a specific object based on a codebook learned from training images without segmentation. Through an unsupervised way, the codebook is learned from cluttered scenes. Given a number of training images containing objects of interest, image patches centered on Harris interest points are extracted and then are separated into compact clusters via agglomerate clustering. The center of each compact cluster is then stored as an entry of the codebook. Given a novel (test) image with extracted patches, a straightforward way is to match them with the codebook entries and the best-matching codebook entries will be used for recognition (e.g. [2]). Alternatively, [75] use a probability voting, activating all entries with similarity larger than a threshold. All the activated locations related to the object center with respect to each entry are saved for voting possible locations of the object center. Mean-shift is then applied in the voting space in order to obtain a promising hypothesis consisting of patches and their surroundings in possible locations. The figure-ground segmentation on an extracted patch from a novel image is thus defined as calculating the probability of labelling each pixel as figure or background, given the learned object hypothesis. The pixel-wise segmentation probability can be obtained by combining segmentation masks of associated patches as in [20]

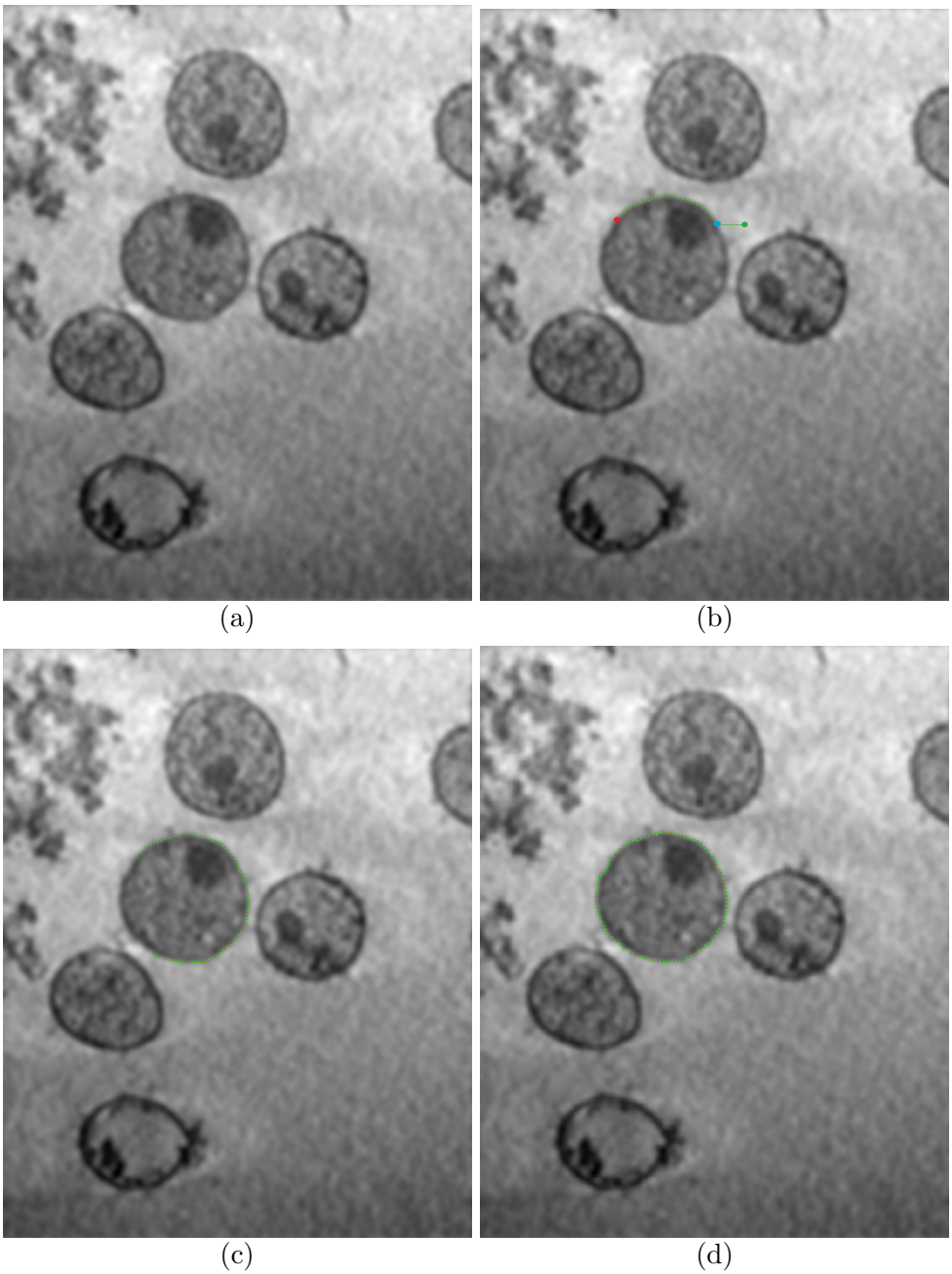


Figure 1.3: A sample procedure of interactive segmentation on a 2D slice of the HIV tomogram using Intelligent Scissor. (a) The original sample slice; (b) one intermediate step; (c) another intermediate step; (d) the final outer surface segmentation of the chosen membrane. The green dotted curves mean contours detected at that moment. To illustrate the procedure in more detail, we manually mark the red, blue and green dots indicating the start point, the end point of the fixed contour and the end point of the interactive contour respectively.

or involving location information as well. The result is a segmentation mask with a pixel-wise confidence map showing the reliability of such segmentation.

The segmentation results may suffer from potential secondary hypotheses covering part of the object. The Minimal Description Length (MDL) principle and an implicit shape model are used in [73] to overcome this problem, where the pixels belonging to the overlap part will be labelled as background with respect to the secondary hypotheses. An extended version of this approach including multiple cues is introduced in [76]. The key is to generate a hypothesis for each cue so as to obtain the basis to integrate these cues. Specifically, it consists of two stages: the first stage ignores cue correlation and generates a codebook and a respective hypothesis for each cue; the second stage reveals cue correlation via fusing cues' probability maps into a common one. Rather than a collection of a partial cover of the most discriminative parts above, the codebook in [20] consists of patches fully covering the object along with their figure-ground labelling.

A number of covers are automatically learned for each object part and each cover can be moved under the constraint of object spatial consistency. In such a way, a larger intra-class variance in shape can be addressed more effectively. It is assumed that the number of covers for a region roughly indicates the likelihood of a region belonging to a figure. Therefore, the likelihood of each pixel belonging to foreground versus background is initialized via the number of covers and then iteratively updated regarding the number of times such pixel occurs respectively. In order to be robust to object deformations while maintain affordable cost of discriminatory power, Bernstein and Amit [13] propose a statistical mixture model for local patches based on photometric invariant edge features. It is assumed that edge features are conditionally independent on each given component. Meanwhile, each component in the mixture model is regarded as a new local feature inheriting the property of photometric invariance. The existence of multiple objects and occlusions can thus be handled through the mixture model learned from a small training set.

Winn and Shotton [137] present a parts-based model incorporating spatial information between connected parts. Arbitrary scaling is thus allowed but its ability to be applied to articulated object categories is not clear. Additional details on top-down segmentation based on codebook can be found in [26], [38] and [72]. Unfortunately, semi-global deformable template matching requires labeled data for supervised learning and is computationally intractable in 3D data with high resolution [7].

**Global deformable template matching.** Another primary branch of deformable template matching is a global model representing the global structure and appearance of given object class ([139], [45], [136], [119]). For instance, LOCUS [136] learns top-down class-specific prior in an unsupervised manner under the assumption that the shape pattern of an object is consistent while the variance in color and texture is limited in terms of a single instance. Thus, given a number of images containing only one target object, it is reasonable to learn the object model – an "average" shape in the form of a global figure-ground mask and a boundary illustrated by a global edge mask. These two masks are designed to well describe the low color/texture variance for a single object while allowing dramatic divergence in intra-class appearance among all the images in training. To generate these two masks, top-down cues (shape and pose) and bottom-up cues (color and edge) are incorporated through a hierarchical generative probabilistic model and the intra-class variances in shape and texture are constrained by applying a smooth deformation field on these cues. In detail, a deformation field and the position and size of an object are sampled followed by applying their corresponding deformation, scaling and translating on the mask and edge image sampled in respect with their prior distributions. The global binary mask is sampled from the transformed mask image whilst the global edge mask is derived from a sampled foreground edge image and a sampled background edge image. These two masks are then used for segmentation via probabilistic inference. Such inference is approximated in an iterative manner. In each iteration, the object's position, size, pose and segmentation are successively refined based on training images. However, the target object in each image should face a constant direction and thus some manual flips are needed. In addition, the assumption that the intra-class variances of color and texture are low may no longer hold, especially in gray level images, and thus derives a weak object model from original images. Again, global deformable template matching requires labeled data for supervised learning and is computationally intractable in 3D data with high resolution.

### 1.3.3 Combining Bottom-Up and Top-Down Segmentation

The state-of-the-art BU segmentation algorithms can produce impressive results due to their ability of being applied to generic images and detect local image discontinuities indicating potential object boundaries. In addition, the computation of low-level cues is straightforward and efficient. However, their major shortcomings are the splitting of semantic meaningful foreground (over-segmentation) and the merging of foreground parts with background (under-segmentation),

due to unavoidable ambiguities that are difficult to be distinguished without additional knowledge about the global structure of the object class. Meanwhile, the contrast between object part and the background is not necessarily strong enough, potentially leading to merging of these two. In contrast, up-to-date TDS algorithms succeed in resolving these BU local ambiguities under the guidance of prior knowledge such as global shape and appearance. However, TDS algorithms have difficulties mainly due to the large intra-class variance in terms of local edges and textures, which limit the extended use of learned representations on general images, and the inefficient matching between the TDS model and the image.

Taken the pros and cons of both BUS and TDS into account, several methods in the literature have therefore suggested means of combining BUS and TDS in an attempt to achieve semantic segmentation in more efficient and effective ways. In fact, the combination of BUS and TDS is observed in various research areas focusing on how the human brain works. Psychophysical and physiological research on the primate visual system has shown that figure-ground semantic segmentation and object recognition interact with each other concurrently in the human vision system (HVS) [107, 24, 101, 109, 109]. In addition, the cooperation between TDS and BUS processes is supported by neurophysiological evidence as well. Depending on the relationships between figure and background, neurons at higher-level of HVS are shown to have an influence on those at low-levels such as visual areas V1 and V2 ([68], [144], [56], [124], [9]). It is observed that the response of many low-level neurons toward the same edge varies depending on the relationships between semantically meaningful foreground and background in an image. In what follows, let's take a review of the state of the art to combine BUS and TDS in either a deterministic or a statistic manner.

**Deterministic Approaches.** One deterministic method of integrating the constraints of both bottom-up and top-down processes is the jigsaw approach ([19]). It poses a binary segmentation as finding an optimal solution of a cost function based on a segmentation tree. From the top of the tree, the input image is split into segments using a coarse-to-fine strategy. Each level represents a segmentation of the image containing segments with respect to different labelled nodes. The bottom-up constraint enforces pixels in homogeneous regions toward the same segment, either foreground or background. Meanwhile, the top-down constraint requires that the segments with respect to foreground should be as close as possible to the initial top-down model. Each node is

a finer segmentation given its parent segmentation. Correspondingly, each local cost function is defined as a linear combination of top-down and bottom-up constraints. The top-down constraint is only applied at the leaf level. It penalizes the dissimilarity between the final segmentation (leaf nodes) and the initial top-down model. Since the top-down labelled model is based on segments of a figure rather than the figure as a whole, this approach is usually called a jigsaw approach. Moreover, the bottom-up constraint is taken into account between two adjacent levels. It encourages the consistency in labelling as its parent if its parent segment is not salient, whereas it tolerates different labels if its parent segment is salient. Hence, the BU constraint penalizes segment where its label is inconsistent with its parent, unless it is a salient region. The sum-product algorithm ([66]) is applied to seek for an optimal labelling minimizing the given full cost function.

To solve the problem of labelling an arbitrary number of objects, Cremers et al. [32] integrate the competition of shape priors into level set segmentation approach. An energy function is generated through a linear combination of shape-based labelling function and level set function. By simultaneously optimizing the level set function and a number of transformation parameters in the energy function through gradient descent, the evolution of contours is enforced by shape priors in selected areas so as to reconstruct familiar shapes. In [119], an integration of elastic shape matching is also discussed given only one shape prior. Given specific object priors, Yu and Shi [138] use a two-layer graph to combine top-down and bottom-up cues. Nodes in one layer are patches derived from top-down object models and edges between them indicate their compatibility. In the other layer, nodes are pixels and the edge between two neighboring pixels implies their similarity (BU cue). A binary segmentation is thus modelled as a hybrid grouping problem: grouping nodes in both pixel-layer and patch-layer into two groups (foreground and background) via the normalized cuts criterion [121]. The resulting optimization is constrained by the association between nodes and patches, represented as edges between these two layers. The eigenvector with the smallest non-trivial eigenvalue is the solution. Additional details on the algorithms of deterministic semantic segmentation can be found in [118, 3, 111, 98].

**Statistical Approaches.** In these approaches, semantic segmentation is formulated as a stochastic optimization problem. The probability distribution of label variable is repeatedly estimated for each pixel. For example, Zhao and Davis [141] directly estimate the probabilities of each pixel belonging to foreground and background through a weighted sum of its respective prob-

abilities in terms of hierarchical template matching and color-based binary segmentation. After applying color-based kernel density estimation and contour-based template matching, the weight is updated with respect to the probability of a person in windows produced by template matching. Then, a new probability of pixel belonging to foreground is generated and the location, size and shape of the windows are adjusted according to the updated foreground. This process repeats until the foreground becomes stable.

Probabilistic graph models have also been widely used for semantic segmentation, such as the Markov Random Field (MRF) model. It incorporates the spatial configuration among neighboring labels as a Markov prior, encouraging the adjacent pixels to be labelled as the same class. For instance, Huang et al. [55] propose a three-layer graphical model integrating bottom-up and top-down cues. In order to tightly couple the MRF-based and the deformable model-based segmentation, a new hidden state representing the underlying contour is added to the bottom of the traditional MRF model. The nodes in the three layers are image pixels, labels of pixels and contour model respectively. Segmentation is thus considered as a joint MAP problem, an estimation of the underlying contour $C$ and region labels $x$ that maximizing a joint posterior on $C$ and $x$ given an image. Since exact inference in this model is intractable, the solution is approximated by decoupling the three-layer model into an extended MRF model and a probabilistic deformable model. Estimation of labels in the extended MRF model is achieved by Belief Propagation (BP) under the contour constraint from the deformable model. The estimated labels in turn contribute to a better estimation of the contour in the probabilistic deformable model based on the variational approaches. In another model named OBJCUT [67], the authors attempted to answer the following three questions: 1) how to make the segmentation conform to figure and background appearance model? 2) how to encourage the segmentation to follow the edges in an image? 3) how to encourage the outline of the segmentation to resemble the shape of the object? In this approach, the top-down shape constraint is involved in figure-ground semantic segmentation via matching an object category model with the given image. As usual, there are two issues in this method: how the top-down model is built and how to integrate it to the segmentation system. Two kinds of objects are taken into account for designing the object category model: non-articulated and articulated objects. The model for the first kind of objects is defined as a set of exemplars (SOE) concerning object shape (boundary) and appearance (texture). This model is learned through a number of manually segmented images

containing the object of interest. In contrast, since large spatial variance should be allowed as well in the case of articulated objects, the model for articulated objects is rather defined as a set of layered pictorial structures (LPS) automatically learned from video sequences, which describe an object as parts in a hierarchical manner and concern not only shape and appearance of each part but also their pairwise configuration. After estimating the pose of the object, a number of samples are obtained from the posterior of the object category model. OBJCUT then relies on an object category specific contrast-dependent random field (CDRF) to model the conditional distribution concerning a unary term, which consists of the appearance potential based on color and the shape potential based on the spatial distance, and a pairwise term consisting of the labelling smoothness prior and a contrast term based on discontinuity. Levin and Weiss [77] propose a fragment-based segmentation on conditional random field that learns to combine bottom-up and top-down cues in a supervised manner. A relatively small set of fragments are learned at first via a feature induction algorithm on candidate fragments and then propagate the segmentation through measuring the image similarity. Specifically, after obtained from an object detector, each path containing part(s) of the object is segmented based on normalized correlation with a number of fragments. A full edge-aligned segmentation is finally produced. This approach requires fewer fragments than pure top-down manner whereas these fragments are local cues, only partially covering the object of interest. Thus it may not only miss object parts but also weaken the propagation efficiency in case of object with significant variance in appearance. In addition, the pairwise configuration between fragments is not considered during matching them with an image. Hence some of the segmented results are not object-like.

Liu et al. [82] proposed another graphical model combining top-down and bottom-up cues in a hybrid manner. In this hybrid graph model, vertices are superpixels while directed and undirected edges are derived from TDS (a codebook) and BUS (mid-level over-segmented regions) respectively. The directed graph (the vertices with the directed edges) is associated with the undirected graph (the vertices with the undirected edges) through a score vector indicating the probability of pixel at hand belonging to each class. The costs of random walk on the directed graph and a minimal cut on the undirected graph are linearly combined to form a new energy function. Hence, the final solution should be a score vector that minimizes this energy function. Additional details on the algorithms of probabilistic semantic segmentation can be found in [118, 3, 110, 140].

Figure 1.4: This example shows the fact that context information plays an important role for human in object recognition, especially when the appearance are not sufficient for recognition.

## 1.4 Using Contextual Cues to Improve Semantic Segmentation

Contextual information exhibits representative configurations of objects, which reduces the search space and is robust to noise. Thus it owns great potential in helping semantic segmentation given the issues mentioned in the previous section. In fact, psychology and vision community have explored the role of contextual information in visual search and recognition for years [6, 15, 104]. Biederman et al. [14] claimed five different types of relations between objects and scenes: support (objects rest on the surfaces of other objects), interposition (objects are surrounded by the background), probability (the possibility of being existed in some scene), familiar size (a limited set of size relations between objects) and position (the possible and impossible locations of an object in a scene given its existence). The first two, often called syntactic relations, reflect the general constraint of gravity and the occlusion due to opaque objects in front of the boundary of another object. The other three, often called semantic relations, are based on object identity. Since semantic relations provide details of interactions among objects in a scene, they are often referred to as contextual features.

To illustrate the role of contextual information in object recognition, one experiment is shown in Fig. 1.4. After capturing an image of the night scene from the roof of a garage (the lower left image), we manually moved a street lamp next to the moon and generated a new image in the lower right corner. Given this new image, subjects described the upper part of the scene as the moon and a star in the sky. Clearly, the local appearances from the respective images (light dots in the first row) are exactly the same and are hence insufficient for recognizing it as a street lamp or a star. Instead, the distance of the relocated street lamp from the moon and the ground makes it be perceived as a star.

We are interested in using contextual information provided by salient objects (easily segmented using appearance features) to segment target objects with low quality based on context features, namely context-sensitive semantic segmentation system. For example, take a look at a $1400 \times 600$ cross-section of a $600 \times 1400 \times 432$ cryo-electron tomogram in Fig. 1.5, which contains both membranes and spikes of nano-scale microvilli. In Fig. 1.5(a), we show four exemplar local windows that potentially includes our target object (spike). In fact, only the two green windows in Fig. 1.5(b) mark the true spikes. Because of the limitation of the imaging devices, the SNR and the contrast are both quite poor in this high-resolution tomogram. Small object segmentation in such noisy and

big data is still an open problem. All traditional semantic segmentation techniques reviewed in this chapter break down when trying to segment the spikes in the nano-scale tomogram. However, at a first glance, the membranes (dark long curves) in Fig. 1.5(b) are more observable (salient) than the spikes and are somewhat distinguishable from the background. Thus, even though it may be difficult to segment the spikes directly, it makes sense to first segment the microvillus membranes and then induce the spikes based on the co-occurrence of membranes and spikes (semantic relation of position). How can we represent the contextual features that describe the relationships between objects and use them for semantic segmentation? Unfortunately, it is not an easy task to organize contextual information in a reasonable framework, which is the main problem this dissertation intends to address.

## 1.5 Summary of Contributions

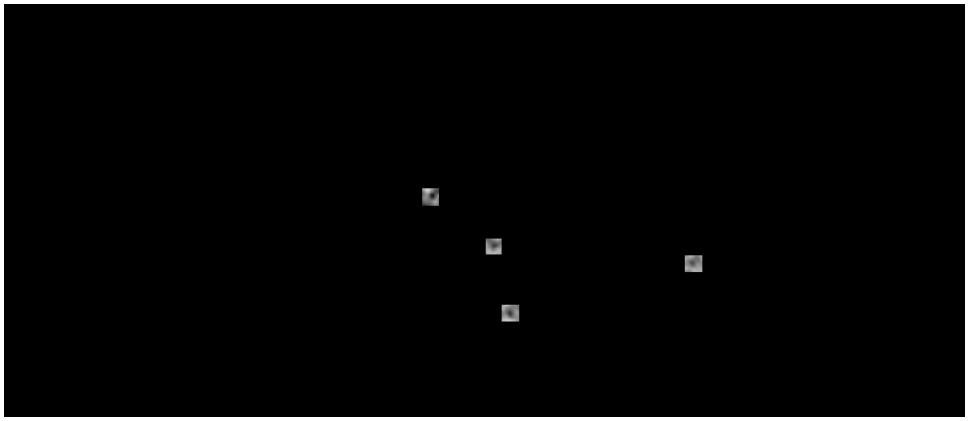
The main contributions of my thesis are as follows:

- We have formulated the problem of context-sensitive semantic segmentation as a well-defined statistic model, proposed a two-stage framework, analyzed its efficiency, and showed how it can be applied to semantic segmentation tasks. In addition, we have developed generalized a context-sensitive algorithm that allows extensive use of features in terms of both appearance and context.

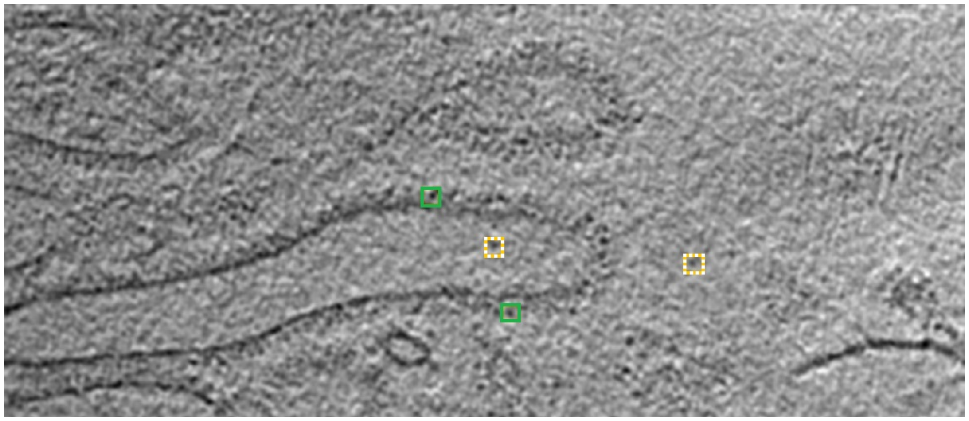
- We have developed two segmentation algorithms of nano-scale membranes in terms of their closeness and varied profile shapes, which is useful for many applications in visualizing plane-like structures of noisy data with high resolution. Further, in related work on surface reconstruction, we have developed an algorithm of reconstructing the semantic surface from 3D light microscopic images.

- We have implemented context-sensitive spike segmentation. Our method is the first algorithm that incorporates context features into nano-scale spike visualization and thus enables automatic spike segmentation. We have also demonstrated excellent performance of our method in the tasks of microvillus and HIV spike segmentation.

- We have applied context-sensitive semantic segmentation to tattoo images. Given the potential tattoo patterns generated from tattoo segmentation, we have demonstrated the state-of-the-art performance in various tasks of tattoo classification.



(a)



(b)

Figure 1.5: This example illustrates a typical context-sensitive semantic segmentation problem. The context information provided by dark and long *context objects* (membranes in (b)) plays an important role for semantic segmentation of small *target objects* (spikes in green windows of (b)) when the SNR is extremely low and thus the appearance of the target objects (as shown in (a)) is not sufficient to distinguish them from the background noise (i.e.: yellow dotted windows of (b)).

## 1.6 Organization

This dissertation is organized as follows. Chapter 2 describes a two-stage statistic framework of context-sensitive semantic segmentation, with theoretical analysis on its potential of outperforming traditional object-centered segmentation methods. Chapter 3 presents the implementation of the first stage of the proposed framework, salient context segmentation, on the problem of automatic microvillus membrane segmentation in cryo-electron tomogram. Chapter 4 extends the first stage to handle semi-automatic but more general segmentation problems in contrast to chapter 3. Chapter 5 formulates multiple context cues and present a hybrid context sensitive model that makes faint spike

segmentation in nano-scale cryo-electron tomogram tractable, with extensive experimental results in exploring the influence of context sensitivity coefficient on semantic segmentation. Chapter 6 tests our framework on natural images for tattoo segmentation and various tasks of tattoo classification. Chapter 7 presents the future works and concludes the dissertation with a summary of contributions.

# CHAPTER 2

# FRAMEWORK

## 2.1 Introduction

As mentioned in the previous chapter, the state-of-the-art methods in the literature are not efficient and practical for segmenting objects in data with high resolution that are only perceptible under the existence of large-scale context objects because of challenging imaging conditions. Thus there is an emerging demand of having a semantic segmentation framework that is general enough to be extensively used for different applications while being efficient enough to minimize its sensitivity towards the scale of data. Based on our analysis on the related works, the segmentation methods with a statistical form is easier to be extensively used than the others. Using the Bayesian rule, we can easily factorize the segmentation "function" so that arbitrary number of useful features can be involved for segmentation. In this chapter, we review the classical statistic semantic segmentation framework, followed by presenting our more general statistical framework of context-sensitive semantic segmentation, along with a discussion on the relationship between the classical framework and our framework.

## 2.2 Classical Semantic Segmentation Framework

Semantic segmentation is aimed at assigning a discrete label $\{o_i\}_{i=1}^{N}$, which takes one of the $K$ values $o_i \in \{1, 2, ..., K\}$, to each of the $N$ basic units in the given data indicating which of the $K$ objects it belongs to. Here the form of the unit can be any region representation of given data, such as pixels, voxels, superpixels, supervoxels, segments, etc. We will use voxels in the rest of this chapter, even though obviously it is not limited to voxels. In a general statistic framework, semantic segmentation is modeled as finding the label $o_i$ that maximizes the following object likelihood function:

$$o_i = \arg\max_{o_i} \Pr(o_i|f_i), \tag{2.1}$$

where $\{f_i\}_{i=1}^{N}$ is a set of features given at each of the $N$ voxels. Here $\Pr(o_i|f_i)$ is the conditional probability density function (PDF) of the presence of the object $o_i$ given a set of features $f_i$.

Consider the fundamental case of semantic segmentation – foreground/background segregation, where foreground is the set of target objects in the given data. Semantic segmentation on multiple object classes can be split into multiple sub-problems of foreground/background segregation. In this fundamental case, $o_i = 1$ when voxel $i$ belongs to the target object, whereas $o_i = 0$ when it belongs to the background. In the classical framework for semantic segmentation, the main source of information derives from local appearance features presented in the target object or its small and primitive spatial neighborhood, such as color, edge, texture and shape. In that sense, objects in the background are assumed to have independent features and are thus considered as distractors, rather than cues, for semantic segmentation. Since this framework is purely based on appearance features of the target, it is often called *object-centered semantic segmentation*. Respectively, the PDF in Eq. 2.1 is re-written as:

$$\Pr(o_i = 1|f_i) \simeq \Pr(o_i = 1|f_i^A), \tag{2.2}$$

where $f_i^A$ is a set of local appearance features on the target object.

Unfortunately, the assumption of the object-centered framework does not often hold in nano-scale. The intrinsic object appearance features $f^A$ are often not distinctive enough for accurate semantic segmentation when SNR is extremely low (see Appendix A for a detailed analysis). Another drawback of the object-centered segmentation framework is its computational cost. Note that $f_i^A$ is a feature set, and every feature needs to be generated through measurements across different locations and scales of the entire volume. Thus the scalability of this framework is intrinsically limited by the large search space.

## 2.3 Context-Sensitive Semantic Segmentation Framework

Due to the problems mentioned above, it is necessary to segment the salient *context objects* in the given data and utilize the possible context cues provided by them to approach the *target object* segmentation. Salient objects are defined as objects that stand out relative to their spatial neighborhoods in an observer's view [21] and thus their appearance feature responses are strong

enough for object-centered segmentation. Instead of modeling the background with context objects as noise, we propose context-sensitive semantic segmentation, a new semantic segmentation framework that is sensitive to context features provided by context objects in the background. In other words, the new framework takes into account the information of detailed interactions among the target object and the context objects in the background. To employ the context features, the problem of semantic segmentation on the target object is re-modeled as two stages: context object segmentation and target object segmentation.

### 2.3.1 Stage One: Model of Context Object Segmentation

Since the SNR is extremely low, even local appearance features of context objects are not distinguishable enough to produce segmentation. Thus it is necessary to employ semi-global features (e.g.: size of connected components, shape model, etc.) in the first stage. Correspondingly, the context object likelihood function in Eq. 2.1 is extended to $\Pr(o_i = 1|f_i^A)$, where $f_i^A$ is the semi-global appearance feature response of context object in voxel $i$, $o_i = 1$ means semi-global context object and 0 otherwise. As a binary segmentation problem, Eq. 2.1 could be replaced by the thresholding strategy for simplicity:

$$o_i = \begin{cases} 1 & \text{if } Pr(o_i = 1|f_i^A) \geq t, \\ 0 & \text{otherwise,} \end{cases} \tag{2.3}$$

where $t$ is a threshold. Note that $f_i^A$ could be any specific object features in any specific problem, relying on which objects provide the context information. This stage ignores context cues. Indeed, it has no other choice since context cues can only be measured relative to the context objects, which are only available after the first stage.

### 2.3.2 Stage Two: Model of Context-Sensitive Target Object Segmentation

Given the hard segmentation of context objects $O = \{o_i\}_{i=1}^N$, it is tractable to compute the context features of target objects. To employ both appearance features ($f_i^{A'}$) and context features ($f_i^{C'}$), the target object segmentation is modeled as finding a discriminant function $\Pr(o_i' = 1|f_i^{A'}, f_i^{C'})$ that predicts the posterior probability of a target object at the $i'th$ voxel given both features. Here $f_i^{A'}$ and $f_i^{C'}$ are used to summarize all types of appearance and context features for target objects at the $i'th$ voxel in the volume, $o_i' = 1$ means target object and 0 otherwise. A factorization could be applied on this conditional probability:

$$
\begin{aligned}
&\Pr(o'_i = 1 | f_i^{A'}, f_i^{C'}) \\
&= \frac{\Pr(o'_i = 1, f_i^{A'}, f_i^{C'})}{\Pr(f_i^{A'}, f_i^{C'})} \\
&= \frac{\Pr(f_i^{A'}) \times \Pr(o'_i = 1 | f_i^{A'}) \times \Pr(f_i^{C'} | o'_i = 1, f_i^{A'})}{\Pr(f_i^{A'}, f_i^{C'})} \\
&= \frac{\Pr(o'_i = 1 | f_i^{A'}) \times \Pr(f_i^{C'} | o'_i = 1)}{\Pr(f_i^{C'} | f_i^{A'})} \\
&\propto \Pr(o'_i = 1 | f_i^{A'}) \times \Pr(f_i^{C'} | o'_i = 1).
\end{aligned}
\tag{2.4}
$$

Note that the context feature response $f_i^{C'}$ fully depends on the world state $o'_i$. Thus the appearance feature response $f_i^{A'}$ in $\Pr(f_i^{C'} | o'_i = 1, f_i^{A'})$ is redundant and is omitted. In addition, as the denominator $\Pr(f_i^{C'} | f_i^{A'})$ of (2.4) is independent of the world state $o'_i$, we only need to consider the numerator $\Pr(o'_i = 1 | f_i^{A'}) \Pr(f_i^{C'} | o'_i = 1)$. The first term of (2.4), $\Pr(o'_i = 1 | f_i^{A'})$, is simply the classical *object-centered model* based on appearance features. The second term, $\Pr(f_i^{C'} | o'_i = 1)$, is a likelihood term that favors context feature responses that are consistent with our prior knowledge about the target. For instance, if it is known that the targets are cars, then the likelihood term will be much larger for road regions than for sea regions. This top-down context cue on known target search is consistent with the discovery that a maximum likelihood strategy is employed for human eye movement to search the most likely locations of the targets.

As a starting point for context sensitive semantic segmentation, we consider the classification of context cues in [46] and thus extend the model into the following form:

$$
f_i^{C'} = \{f_i^{C_{se}}, f_i^{C_{sp}}, f_i^{C_{sc}}\}, \tag{2.5}
$$

where $f_i^{C_{se}}$, $f_i^{C_{sp}}$ and $f_i^{C_{sc}}$ are the semantic context (e.g: probability of coexistence), the spatial context (e.g.: position and orientation) and the scale context (e.g.: size) of the target object with respect to another nearby large-scale object respectively. Hence Eq. 2.4 can be decomposed into four terms, the last three of which take an additional type of context information into account sequentially:

$$
\begin{aligned}
&\Pr(o'_i = 1 | f_i^{A'}, f_i^{C'}) \\
&\propto \Pr(o'_i = 1 | f_i^{A'}) \times \Pr(f_i^{C'} | o'_i = 1) \\
&= \Pr(o'_i = 1 | f_i^{A'}) \times \Pr(f_i^{C_{se}}, f_i^{C_{sp}}, f_i^{C_{sc}} | o'_i = 1) \\
&= \Pr(o'_i = 1 | f_i^{A'}) \times \Pr(f_i^{C_{se}} | o'_i = 1) \times \Pr(f_i^{C_{sc}} | f_i^{C_{se}}, o'_i = 1) \times \Pr(f_i^{C_{sp}} | f_i^{C_{sc}}, f_i^{C_{se}}, o'_i = 1).
\end{aligned}
\tag{2.6}
$$

In our work, we focus on how to utilize different types of context information to gradually not only improve the accuracy but also significantly accelerate semantic segmentation on a large-scale volume.

## 2.4 Information Theoretical Analysis

Both the classical object-centered framework and our context-sensitive framework can be concerned from the view of information theory. As logarithm is a monotonically increasing function, we can re-write the object-centered framework (Eq. 2.2) into the following log probability:

$$
\log \Pr(o_i = 1 | f_i^A) \simeq \underbrace{\log \Pr(f_i^A | o_i = 1)}_{\text{Log-likelihood}} - \underbrace{\log \Pr(f_i^A)}_{\text{Self-information}} + \underbrace{\log \Pr(o_i = 1)}_{\text{Location prior}}. \tag{2.7}
$$

The first term on the right side, $\log \Pr(f_i^A | o_i = 1)$, is a log-likelihood that reflects how likely it is to observe the feature response of $f_i^A$ in the presence of the target object. Thus it encodes the top-down prior knowledge of the target object. For instance, if we know that our task is to segment green apples from an image, then the log-likelihood of a green pixel will be much higher than that of a blue pixel.

The second term in Eq. 2.7, $-\log \Pr(f_i^A)$, does not include the label variable $o_i$ and is hence independent of any prior knowledge of the target object. In information theory, it is known as the *self-information* of the appearance feature $f^A$, which increases when the probability of the appearance features $f^A$ decreases. It thus implies that rarer features are more informative for semantic segmentation. Returning to our example of segmenting green apples, even if we can make a strong assumption that the target apple is absolutely green ($\Pr(f^A = green | o_i = 1) = 1$), this information still become helpless when everything in this world is green. Conversely, the color in such case even makes it more difficult to segment the target from the background. Therefore, appearance features of the target object are more useful when they are relatively rare in the

background. Note that this term is purely data-driven, it is also called bottom-up saliency in the literature of vision science.

The third term in Eq. 2.7, $\log \Pr(o_i = 1)$, depends only on the object label $o_i$ at location $i$ of the given data and is thus independent of the appearance features $f^A$. It biases the prior knowledge of object location and thus favors the one that is most likely to appear in current location. It is usually assumed to be uniform over the possible object labels.

As the third term contributes equally to each class of objects when assumed uniform, it is often omitted from Eq. 2.7. Then we can re-write this equation into the following form:

$$\begin{aligned}
&\log \Pr(o_i = 1 | f_i^A) \\
&\simeq \underbrace{\log \Pr(f_i^A | o_i = 1)}_{\substack{\text{Top-down} \\ \text{prior}}} - \underbrace{\log \Pr(f_i^A)}_{\substack{\text{Bottom-up} \\ \text{saliency}}} \\
&= \log \frac{\Pr(f_i^A | o_i = 1)}{\Pr(f_i^A)} \\
&= \underbrace{\log \frac{\Pr(f_i^A, o_i = 1)}{\Pr(f_i^A) \Pr(o_i = 1)}}_{\substack{\text{Pointwise} \\ \text{mutual information}}}.
\end{aligned} \tag{2.8}$$

In information theory, this new equation formulates the classical object-centered framework as the *pointwise mutual information* between the appearance features and the presence of a target object. Therefore, it intuitively favors the appearance feature values that are more usual in the presence of the target object rather than in the absence of the target object. Returning to the example of segmenting green apples, if everything in this world is green, green is then a poor appearance feature because of its presence in both green apples and the other objects.

Our context-sensitive semantic segmentation framework can be analyzed in a similar manner. As the first stage of our context-sensitive segmentation framework is classical object-centered framework, we will focus on analysis of the second stage. The log probability of the second stage in our context-sensitive framework (Eq. 2.4) is:

$$\begin{aligned}
&\log \Pr(o'_i = 1 | f_i^{A'}, f_i^{C'}) \\
&\simeq \log \Pr(o'_i = 1, f_i^{A'}, f_i^{C'}) - \log \Pr(f_i^{A'}, f_i^{C'}) \\
&= \log \Pr(o'_i = 1) + \log \Pr(f_i^{A'} | o'_i = 1) + \log \Pr(f_i^{C'} | o'_i = 1, f_i^{A'}) - \log \Pr(f_i^{A'}, f_i^{C'}).
\end{aligned} \tag{2.9}$$

Based on the previous assumption that appearance features $f^A$ and context features $f^C$ are conditionally independent given the target, we can re-write Eq. 2.9 into the following form:

$$\begin{aligned}
&\log \Pr(o'_i = 1 | f_i^{A'}, f_i^{C'}) \\
&\simeq \underbrace{\log \Pr(f_i^{A'} | o'_i = 1) + \log \Pr(f_i^{C'} | o'_i = 1)}_{\text{Log-likelihood}} - \underbrace{\log \Pr(f_i^{A'}, f_i^{C'})}_{\text{Self-information}} + \underbrace{\log \Pr(o'_i = 1)}_{\text{Location prior}}.
\end{aligned} \tag{2.10}$$

Again, the first term on the right side, $\log \Pr(f_i^{A'} | o'_i = 1)$, is the log-likelihood that reflects how likely it is to observe the response of appearance feature $f_i^{A'}$ in the presence of the target object. Similarly, the second term, $\log \Pr(f_i^{C'} | o'_i = 1)$, is the log-likelihood that reflects the probability of observing the response of context feature $f_i^{C'}$ in the presence of the target object. Recalling our task of segmenting green apples from an image, the log-likelihood of finding a green apple on a tree will be much higher than that of finding it in the sky. These two terms encode the top-down prior knowledge of the target object.

The third term in Eq. 2.10, $-\log \Pr(f_i^{A'}, f_i^{C'})$, does not include the label variable $o'_i$ and is hence independent of any prior knowledge of the target object. It is the *self-information* of all the available features (both $f_i^{A'}$ and $f_i^{C'}$), which increases when the joint probability of all the available features decreases. It also implies that rarer features are more informative for semantic segmentation.

The forth term in Eq. 2.10, $\log \Pr(o'_i = 1)$, still depends only on the object label $o_i$ and is thus independent of any feature. Again, it biases the prior knowledge of objects and thus favors the one that is most likely to appear. It is usually assumed to be uniform over the possible object labels and is often omitted from Eq. 2.10. Then we can re-write this equation into the following form:

$$\begin{aligned}
&\log \Pr(o'_i = 1 | f_i^{A'}, f_i^{C'}) \\
&\simeq \log \Pr(f_i^{A'} | o'_i = 1) + \log \Pr(f_i^{C'} | o'_i = 1) - \log \Pr(f_i^{A'}, f_i^{C'}) \\
&= \underbrace{\log \frac{\Pr(f_i^{A'}, o'_i = 1)}{\Pr(f_i^{A'}) \Pr(o'_i = 1)}}_{\substack{\text{Original pointwise} \\ \text{mutual information}}} + \underbrace{\log \frac{\Pr(f_i^{C'}, o'_i = 1)}{\Pr(f_i^{C'}) \Pr(o'_i = 1)} - \log \frac{\Pr(f_i^{A'}, f_i^{C'})}{\Pr(f_i^{A'}) \Pr(f_i^{C'})}}_{\substack{\text{Additional pointwise} \\ \text{mutual information}}}.
\end{aligned} \tag{2.11}$$

In information theory, this new equation consists of three terms of *pointwise mutual information*. Similar to the object-centered segmentation framework, the first term is the point-wise mutual

information between the appearance features and the presence of a target object. It intuitively favors the appearance feature values that are more usual in the presence of the target object rather than in the absence of the target object. The second and third term together provide additional information in semantic segmentation. The second term is the point-wise mutual information between the context features and the presence of a target object, whereas the third term concerns the point-wise mutual information between the appearance features and the context features of a target object. Therefore, the second term favors the context feature values that are more usual in the presence of the target object rather than in the absence of the target object, whereas the third term penalizes the correlation between appearance features and context features used for segmentation.

## 2.5 Summary

In this chapter, we reviewed the classical semantic segmentation framework at first. As this framework is not efficient for segmenting objects that are only perceptible under the existence of large-scale context objects, we then proposed a novel two-stage statistical framework for context-sensitive semantic segmentation. By analyzing them in the view of information theory, additional information employed by our context-sensitive model is explicitly presented. Varied features are allowed to be used in our framework because of its statistic formulation. In the next three chapters, we will apply this framework in the problem of nano-scale spike segmentation in cryo-electron tomogram and tattoo segmentation in more natural images, showing the merits of our context-sensitive semantic segmentation framework in contrast to the traditional framework of object-centered semantic segmentation.

# CHAPTER 3

# 3D SALIENT CONTEXT OBJECT SEGMENTATION ON NANO-SCALE

## 3.1 Introduction

As mentioned in the previous chapter, our method consists of a number of steps that can be grouped into basically two sequential stages: salient context object segmentation and context-sensitive faint target segmentation. This chapter explores the implementation of the first stage on one difficult situation – the data imaged at nano scale.

Nano-scale imaging technologies have been commonly used for visualizing in three-dimensions (3D) the structures of size less than 100 nanometers (i.e.: molecules, proteins, viruses, etc.). Thus they benefit a wide range of applications such as biophysics, biochemistry, material science, environmental technologies, micro-processor manufacturing and medicine [113, 142]. For example, the top image of Fig. 3.1 shows a slice from a volumetric image (tomogram) of microvilli, acquired by 3D cryo-electron microscopy [87]. For visualization purpose, it is the low-passed filtered image. Such data is critical for biophysicists, as it provides the access to the internal organization of the microvilli at an unprecedented detail that is possible for the identification and the quantitative analysis of spikes, which are the workhorse of protein production. The nano-scale imaging allows us to investigate structures that are very close to their native states and potential spatial relations between them.

Generally, nano-scale structure studies rely on several critical stages: imaging process, segmentation, 3D classification, 3D reconstruction, and statistical analysis [42]. Among these stages, segmentation is of utmost importance. From an image processing perspective, this task aims at extracting target structural components in nanometers from a volumetric image by labeling the voxels that compose them. The common and natural way of nano-scale semantic segmentation is carried out manually, using some visualization tools such as IMOD [65] and Fiji [117]. This is not only subjective but also labor-expensive, especially when the rapid advances in automation of nano-scale imaging have led to a dramatic increase in the speed of data collection. Therefore, the

Figure 3.1: In this chapter, we address the task of context object (in our example, membrane) segmentation on a 3D tomogram, which is the first stage of our context-sensitive semantic segmentation. An exemplar "slice" of a 3D cry-electron tomogram is shown in the top image. One sample membrane is marked by a green curve in the bottom image.

growing amount of human effort required in segmentation becomes the bottleneck of nano-scale research.

There exists a large number of segmentation algorithms in the computer vision literature that attempt to automate the segmentation process, represented by watershed [130], active contour [17, 8], level set [31], sliding window [33], graphcut [23], normalized cut [121] and Gaussian Mixture Model (GMM) [90]. However, these algorithms achieve limited success in nano-scale data [113, 42]. The task of automation on nano-scale segmentation is primarily hindered by the coexistence of two problems: the low signal-to-noise (SNR) ratio and the large scale. The first problem is intrinsic

to nano-scale imaging because of how the image is produced. In an attempt to image nano-scale objects, it is necessary to use enough doses of electrons to capture measurable contrast. On the other hand, an increase in the use of electron does tend to damage the structure of nano-scale objects. Based on this trade-off, it is common to observe nano-scale data with low SNR and low contrast (as shown in Fig. 3.1). Most of the state-of-the-art in segmentation assumes relatively high SNR and thus all fail on nano-scale data. One way to alleviate this problem is to apply smoothing before segmentation [108, 131]. But it also reduces the resolution of edges and features. Thus it sacrifices the accuracy of semantic segmentation, which is especially important for further analysis on fine-scale interests. Traditional anisotropic filters [16, 59, 43, 44] attempt to inhibit noise while preserving edges. However, these strategies also do not work well under the exsistence of extremely low SNR. The second problem, the large scale, derives from the first problem. Due to the low SNR, the extremely fine-scale structures (such as the spikes with magenta labels in Fig. 1.5 (b)) can only be distinguished from the cluttered background under the existence of some larger-scale context cues (such as the membrane marked by a green curve in the bottom image of Fig. 3.1). Thus it is necessary to capture both larger-scale and fine-scale structural components in the data, which requires high resolution (more than 360 million voxels). Consequently, all the methods mentioned above, based on sophisticated operations, are intractable and inapplicable concerning the size of the nano-scale data.

Since the existing methods are inefficient, we need to design a novel segmentation algorithm for nano-scale data. It should capture the fine-scale structures in big data under extremely low SNR, and voxel-wise segmentation must be accurate and efficient. Indeed, faint nano-scale objects can only be distinguished from the cluttered background under the existence of nearby salient objects. Thus salient object segmentation in nano-scale data is inevitable for producing the context cues that aid nano-scale object segmentation. To address these concerns, we have developed a spike segmentation algorithm on nano-scale tomogram, as an application showing the advantages of our framework in the previous chapter. In what follows, we talk about the salient context segmentation, using membrane segmentation as an exemplar and, in the next chapter, about context-sensitive spike segmentation using context cues.

## 3.2 Algorithm

Based on the prior knowledge from data collection, voxels with high intensity value have great potential of belonging to a nano-scale object rather than the background. The assumption of our framework is that the appearance features for larger-scale context object are distinguishable enough for segmentation despite the extremely low SNR. Therefore, our first stage is to segment the larger-scale objects (membranes) that supply strong contextual constraints to the fine-scale faint targets (spikes). This stage comprises the following steps: scale selection, context object segmentation based on generative model, thresholding and globalization.

### 3.2.1 Scale Space

Based on scale space theory in discrete signals, features can be segregated according to the scale [80]. At a given scale, the features sized larger than or equal to this scale are preserved, whereas the other features are filtered out. Since SNR is extremely low in our case, the context objects are seriously corrupted by the noise in the finest scale (the original input). Thus the appearance features in the finest scale are ambiguous for segmentation. Our first step is to explore the scale space and identify the appropriate scale, in which noise on the context objects is mostly inhibited whereas the edge responses of the context objects are almost preserved. To simplify the details and thus emphasize our framework, we just applied context segmentation in 2D on a slice-by-slice basis and stacked the segmented contours in 3D. This is also reasonable for nano-scale data because the resolution in its z-direction is reduced due to the missing wedge effect [60]. More sophisticated 2D/3D large-scale object detectors under extremely low SNR [47, 95, 96] could be used in practice.

To achieve such purpose, we first create a Gaussian scale space on the original slice $I$ and manually select scale $k$, for which noise on the membrane is filtered out in the respective scale image $S_k$. Mathematically speaking, $S_{j+1} = S_j * G_\sigma$ such that $j = 1, 2, ..., k$, $S_1 = I$, $*$ means convolution, $S_j$ is the scale image in the $j$'th scale and $G_\sigma$ is the Gaussian filter of certain standard deviation $\sigma$. Here we assume that membrane features share the similar thickness (approximately 3 voxels) in our case and thus only the scale $k$ is desired for membrane segmentation. For context object features with different scales, the appropriate scales need to be explored for remaining procedures. In practice, this step is replaced by a convolution of tomogram with a Gaussian filter whose standard deviation is close to the sum of the thickness of the membrane.

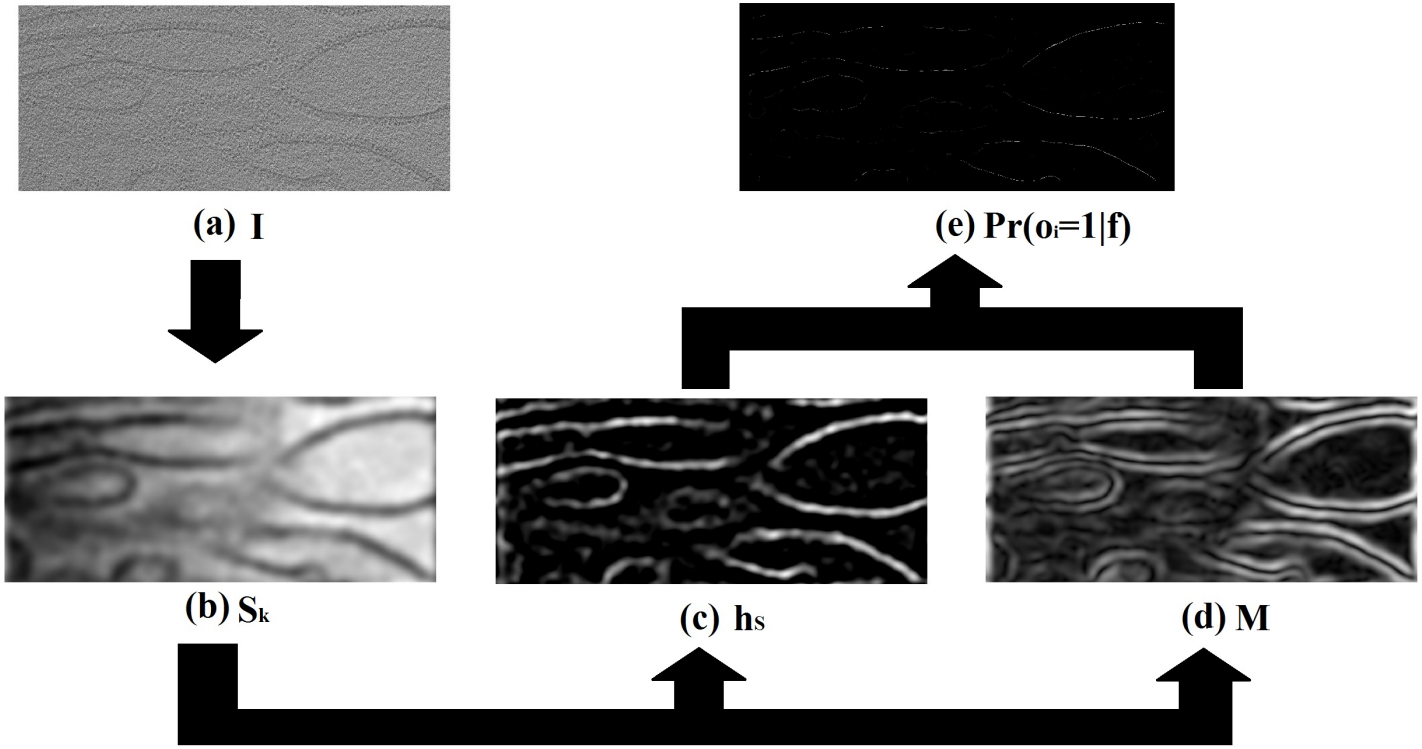



Figure 3.2: Illustration of membrane segmentation steps on an exemplar 2D slice.

Figure 3.3: A linear Difference-of-Gaussian (DoG) filter that models the off-center/on-surround receptive field.

### 3.2.2 Context Object Likelihood Channel

**Membrane Model.** Due to extremely low SNR, the context objects are seriously blurred in the selected scale that inhibits most of noise. Since the targets are assumed to be dark on a bright background in nano-scale data because of the means of imaging, we use an off-center/on-surround receptive field model to 'activate' the regions corresponding to the smoothed target. More precisely, given the thickness of the membrane, we applied a 3D linear Difference-of-Gaussian (DoG) filter $f_{d_i,d_o}$ with the diameter of the inner Gaussian $d_i$ close to the thickness of the membrane on the scale image $S_k$ (the central 2D slice cut of which is illustrated in Fig. 3.3). A half-wave rectifier function $R(x) = max(0, x)$ was then applied on the resultant feature response map to produce an initial map $h_s$ indicating the regions of smoothed membrane:

$$h_s = R(f_{d_i,d_o} * S_k) = \begin{cases} f_{d_i,d_o} * S_k & \text{, if } f_{d_i,d_o} * S_k > 0, \\ 0 & \text{, otherwise.} \end{cases} \tag{3.1}$$

**Membrane Likelihood Channel.** So far we have enhanced the connectedness of the membrane at the sacrifice of their local contrast. To recover the contrast for accurate segmentation, we observed that the gradient strength map of the smoothed image $M = \nabla_x {S_k}^2 + \nabla_y {S_k}^2$ itself carries more accurate contour information of the membrane. More precisely, the center (ridge) of the membrane is extended into a smooth region in $M$, showing as a local minimum in the profile of gradient values across the membrane. Meanwhile, the two sides of the membrane on such profile are shown as two peaks. We thus combine the initial membrane map with the gradient strength map, deriving the final membrane-likelihood

$$\Pr(o_i = 1|f^A) = \|\Upsilon(\frac{h_s}{M+\varepsilon})\|, \tag{3.2}$$

where $i = 1, ..., N$ where $N$ is the number of voxels in the tomogram, $\|.\|$ indicates the normalization operator that rescales the values to the range $[0, 1]$, $\Upsilon(x)$ means non-maximum suppression (NMS) on input image $x$, and $\varepsilon > 0$ is a sufficiently small scalar to avoid dividing by zero. Figure 3.2 shows the exemplar region in original image $I$ with its corresponding $S_k$, $h_s$, $M$ and $\Pr(o_i = 1|f^A)$.

### 3.2.3 3D Thresholding and Globalization

Based on the membrane-likelihood channel of all the slices as local features, we are able to develop semi-global features for membrane segmentation based on its connectedness. Since contextual objects are the largest objects in the given data, the size deriving from threshold is a reasonable semi-global feature for segmentation. For simplicity, we just use a threshold $t$ to control the segmentation. A voxel will be marked as the membrane if its respective membrane-likelihood is larger than the threshold $t$. Similar to the last step of Canny edge detector [25], we can replace the single thresholding strategy by first adopting a high threshold $t_h$ on the 3D membrane-likelihood channel $g_m$ and producing the largest $N$ 3D connected components $\{C_i|i = 1, 2, ..., N\}$ as seed voxels. Then a lower threshold $t_l$ was also applied on $g_m$. The connected components overlapping any one of $\{C_i\}$ are labelled as membranes, denoted $\{M_k|k = 1, 2, ..., N'\}$ and $N' \leqslant N$. As the context objects appear to be large structures in our case, the largest $K$ connected components in 3D are marked as the final segmentation. The choice of $K$ depends on not only the number of membranes in the given tomogram but also how many of them break into separate pieces because of missing wedge effects. More sophisticated and efficient hysteresis thresholding strategies can be found in [95].

## 3.3 Experiment

### 3.3.1 Dataset and Experimental Setup

For the experiment, the tomogram is acquired using a $600 \times 1400 \times 432$ cryo-electron microscope, where the SNR and the contrast are very low. Microvillus membranes are the most salient objects in the tomogram. The $z$ axis is the direction parallel to the electron beam, along which the missing wedge effect gives rise to a loss of the resolution. Hence we present the segmentation result in 2D slices along the $z$ axis for the visualization purpose at first. In our tomogram, the spikes are densely

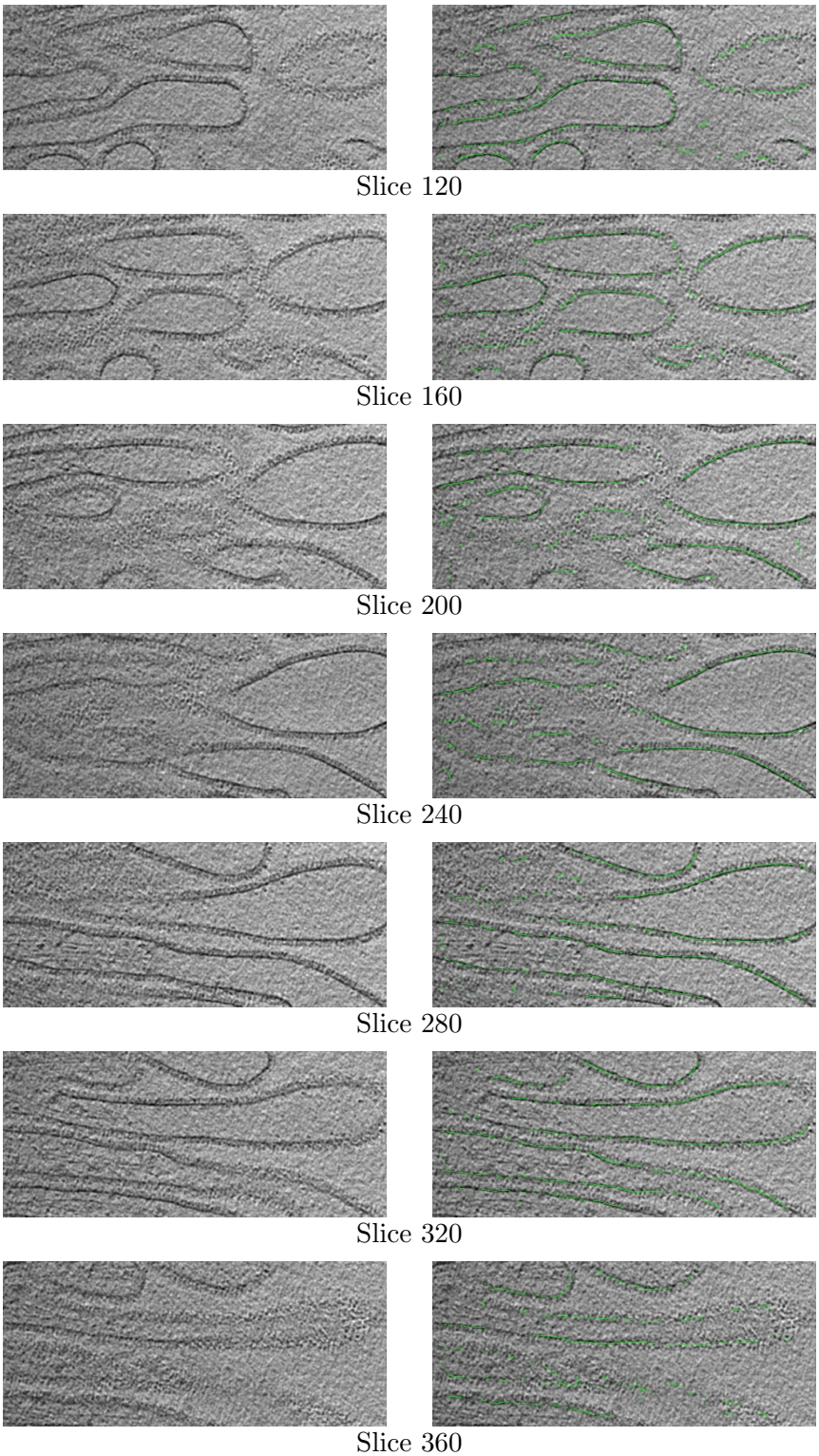



Figure 3.4: Visualization of microvillus membrane segmentation in 2D view. The first column is the original slice, whereas the second is the respective segmentation with membranes marked by green curves.

arrayed on the outer surface of the membrane. Hence the contour of the membrane and its spikes as a whole presents a relatively clear contrast. Hence, we set the standard deviation in the scale-space step as the sum of the general thickness of the microvillus spikes and the thickness of the membrane, $\sigma = 20$. A sample slice of the resultant 3D scale image $S_k$ is shown in Fig. 3.2(b). Meanwhile, given another prior knowledge that the general thickness of the microvillus membrane is 3, we set the diameter of the inner Gaussian $d_i = 3$ and the one for the outer Gaussian $d_o = 2 \times d_i$. A sample slice of the resultant 3D membrane likelihood channel $\Pr(o_i = 1|f^A)$ is shown in Fig. 3.2(e), with two terms in its division $h_s$ and $M$ shown in Fig. 3.2(c) and Fig. 3.2(d) respectively. The threshold $t$ is set to preserve $0.004 \times N$ voxels, where $N$ again is the number of tomogram voxels. This threshold allows us to control the number of false positive based our visual estimation of the number of the membrane voxels in the given tomogram. Finally, the number of marked connected components is set as $K = 10$ to produce the final segmentation.

### 3.3.2 Visualization of Segmentation Result in 2D Slices

In Fig. 3.4, we sample a number of 2D slices from the microvillus tomogram to show our segmentation result of the largest membrane, which is presented as the largest ellipse-like contour that is closest to the right boundary of each slice. The first column of Fig. 3.4 shows the original slices (low-filtered by a 3D Gaussian with standard deviation $\sigma = 3$ for visualization purpose), whereas the second column shows the respective membranes marked by green curves. It shows that our segmentation manages to extract the desired contour of microvillus membranes, regardless of the low SNR and the low contrast. Since our segmentation is carried out on the entire 3D tomogram, segmentation of other membranes may also show up in these sample slices.

We can clearly observe the missing wedge effect on the largest membrane from the first column of slice 120 and slice 360 in Fig. 3.4. As mentioned in our previous analysis, missing wedge effect gives rise to loss of contrast along the $z$ axis. Because the local surface of the largest membrane is orthogonal to the electron beam in these two slices, the contrast of the membrane surface is extremely poor. In contrast, the spikes arrayed on this membrane grow in the direction that is orthogonal to local membrane surface. Thus the missing wedge effect on them is minor. Consequently, in these two slices, we observe a large number of spikes (appear as small dark dots) grouped as a shape that roughly matches the shape of the respective membrane boundary in nearby respective slice 160 and slice 320. As shown in the second column of slice 120 and slice 360, our method is sensitive

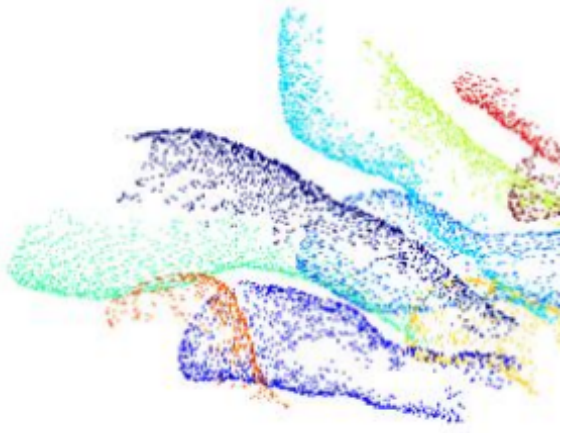


Figure 3.5: Visualization of segmentation result in 3D space.

to the missing wedge effect. On one hand, it considers the boundaries shaped by the spikes as the boundary of the membrane, which produces false positives for segmentation. On the other hand, it somewhat preserves the shape of the membrane and thus benefits the observation of the membrane structure with limited information.

### 3.3.3 Visualization of Segmentation Result in 3D Space

Figure 3.5 is a screen shot of the 3D view of the largest 10 segments (connected components) of the final membrane segmentation, centered at the largest membrane for visualization purpose. Each connected component in 3D space is marked with one color. Thus the spatial relationships among different membranes are well preserved for the connected component analysis. It shows clearly that the 2D contours in Fig. 3.4 are connected throughout the tomogram to represent the voxels belonging to the same membrane piece as a whole. Note that the result is in 3D and they can thus be visualized from different view angles. Another valuable observation in 3D space is a 3D view of the missing wedge effect. It is easy to observe that the largest two membranes are separated into two pieces because of missing information on two sides along their ridges. For example, the largest membrane is separated into two pieces with spring green and midnight blue respectively.

With the segmented outer surface of each membrane, the normal of the surface on each voxel can be computed easily and then provides potential context cues of spikes arrayed on the outer surface.

Clearly the segmentation allows systematic study of nano-scale membrane in noisy tomogram with high resolution and makes large scale studies feasible.

## 3.4 Summary

As one of the most critical steps in analysis of data captured by nano-scale imaging, the segmentation of nano-scale objects is critical for researchers to investigate their structures and functions. In this chapter, we have developed a nano-scale membrane segmentation technology that allows visualization of microvillus membranes at nanometer resolutions in 3D. Thus it satisfies the intrinsic demand on designing explicit models that succeed in extracting implicit nano-structures. By utilizing the power of the membrane model inspired by receptive field in human visual system, we managed to overcome the intrinsic difficulties of nano-scale imaging deriving from low SNR, low contrast and large data scale. In contrast, existing segmentation methods, including commonly used manual segmentation and computer vision algorithms developed for segmentation, often fail on nano-scale data.

Our experimental results allow us to illustrate the missing wedge effect in a straightforward manner. The loss of information along the direction that is parallel to the electron beam is easy to be observed in both 2D and 3D view of our segmentation results. Thus it may provide a tool for the researchers to observe and/or estimate the missing wedge effect on a given data by visually comparing the extracted structure and the expected structure.

In this chapter, we have developed the first stage of our context-sensitive framework, assumed that the membrane is ridge-like. As the texture inside the microvillus membranes is somewhat homogeneous, this assumption works well. However, it is more often that various types of structures or tissues may appear inside the membrane. Despite the ridge-like shape of the membrane in reality, its profile may partially appears as other shape in the data because of other inside structures attached on the membrane surface. Thus there is a natural demand in the first stage of our framework to answer how to deal with membranes appeared in varied shapes. In addition, the prior knowledge of microvillus structures allows us to assume that only context objects (membranes) and small target objects (spikes) are supposed to appear in cryo-electron tomograms. In a wider range of tomograms, it is more than likely that the context objects are not the only salient objects in the given data. In such case, it is no longer reasonable to consider the salient object segmentation in

the first stage of our framework as the context object. How to identify the context objects that provide context cues for the target objects is also a non-trivial question. To answer these two questions which hinder the extensive use of our method proposed in this chapter, we will propose an alternative and more general solution in the next chapter before moving to demonstrate the second stage of our framework.

# CHAPTER 4

# INTERACTIVE SEGMENTATION OF CONTEXT OBJECT IN 3D SPACE

## 4.1 Introduction

In the previous chapter, the context object segmentation engineers a local appearance model of the context object, fits this mathematical model to the given data, and thresholds on the posterior probability of voxels belonging to the context object (describing how well they fit the proposed context object model). As discussed in the end of the previous chapter, it is based on two assumptions that may limit its extensive use in other context-sensitive semantic segmentation problem. However, it is often difficult to find a general mathematical model that appropriately describes objects with context information.

First of all, such difficulty arises from the fact that cluttered background may consist of noise objects from other categories. As noise objects are usually indistinguishable from the context objects in terms of local appearances, they may also yield strong feature responses of model fitting. Hence, the presence of noise objects results in a large number of false positives. Take Fig. 4.1 for instance, the circle-like or ellipsoid-like salient structures are membranes of human immunodeficiency viruses (HIV's) and we are interested in segmenting the spikes arrayed on the HIV membranes. In contrast to the microvillus tomogram in the previous chapter, the existence of structures inside the membrane may change the intensity values along the profile that is perpendicular to the membrane surface. The green profile in Fig. 4.1, similar to the profile of microvilli tomogram, is ridge-like as its density value decreases as a distance function in term of the center of the membrane. Instead, the red and blue profiles appear to be edge-like as the intensity values on one side of the membrane are similar to that of the membrane. Figure 4.2 shows these three profiles in terms of the intensity values with the corresponding colors. There is a number of work in the literature attempting to model different types of membranes [116, 94]. Similarly, methods have been proposed for segmenting other specific nano-scale structures, such as micro-tubules [133] and filaments [115, 135].

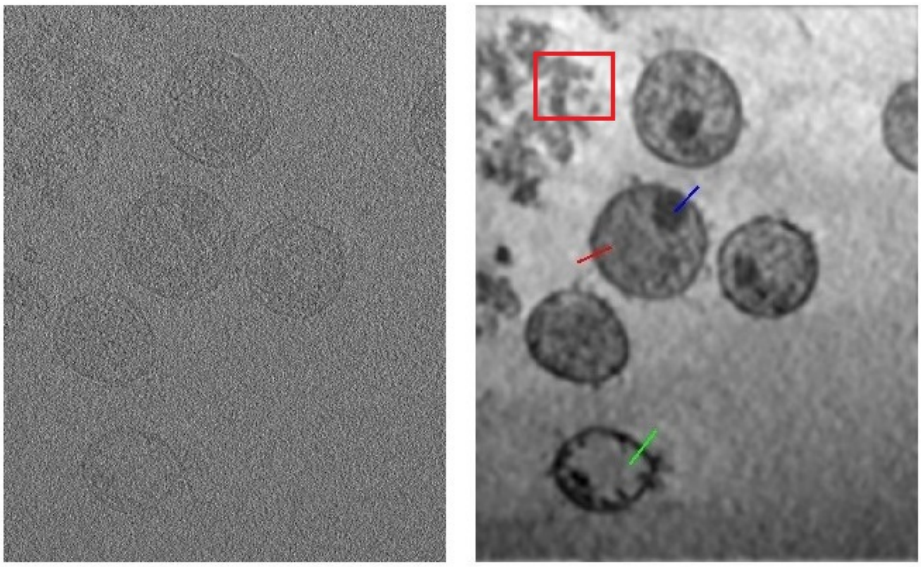


Figure 4.1: This example illustrates the existence of cluttered background both inside and outside the context object. The left image is the original 2D exemplar slice from a 3D tomogram of HIV. The extremely low SNR hinders us in observing many objects in this slice. Thus, for visualization purpose, on the right is a low-pass filtered image of the original slice using a 2D Gaussian filter with variance $\sigma = 5$.

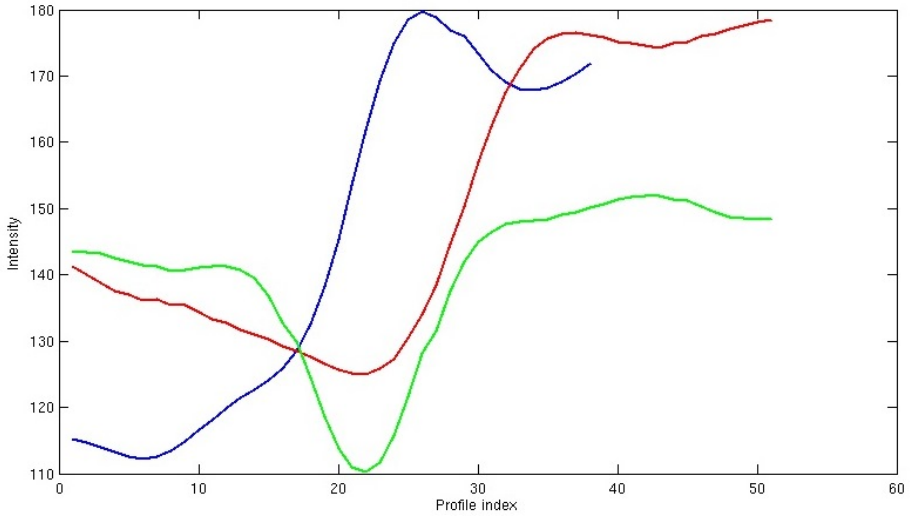



Figure 4.2: This example illustrates the intensity values on each profile of Fig. 4.1 with the corresponding color. The profile index increases from the inside end point to the outside end point.

However, due to the complexity of nano-scale structures, these methods are hard to be used in the general case (i.e. for all types of nano-scale structures).

In addition, the context objects are often not the only large scale objects in the given data. The size of a region in such case is no longer a good semi-global feature for refining the segmentation result of local appearance models. Even worse, if the shape of the context object is irregular, it is also difficult in designing a shape model as global appearance model for segmentation. For example, the structures inside the red box of Fig. 4.1 are a group of noise objects in the background appearing as a salient object. The surfaces of these structures are also edge-like. Thus localization of the object we are interested in becomes a more challenging problem. For visualization purpose, the slice images listed in the rest of this chapter are all low-pass filtered if not specified otherwise.

Taken the difficulty of designing specific object features into account, we need a more general and applicable means of object localization for further segmentation. This is a basic problem of foreground/background segmentation. A typical way of efficiently approaching this task is to involve 2D interactive tools, such as Magic Wand in Adobe Photoshop 7 using texture information, Intelligent Scissors [99] using edge information, Bayes matting [99] using color distributions, or Graph Cut [49, 22, 23] combining both textures and edges. The difficulty then lies in how to extend the result of 2D interactive segmentation into 3D accurately and smoothly. In this chapter, two interactive foreground/background segmentation algorithms were proposed to overcome such difficulties. The first algorithm was used to solve a novel problem of Drosophila (fruit-fly) head segmentation in 3D microscopic images. The second one was used for segmenting HIV membranes for further analysis on HIV spikes.

## 4.2 Interactive Segmentation of Drosophila Head in 3D Space

### 4.2.1 Motivation

As drosophila is widely used as a model for human diseases (e.g. [88]) and has a relatively rapid generation time, it is an ideal species for testing phenomic approaches. This leads to the requirements of efficient acquisition and modeling of three dimensional parts of drosophila. However, the acquisition and modeling present unique computational and algorithmic challenges. For example, while a typical drosophila is about two millimeters in length, it has very complicated forms and

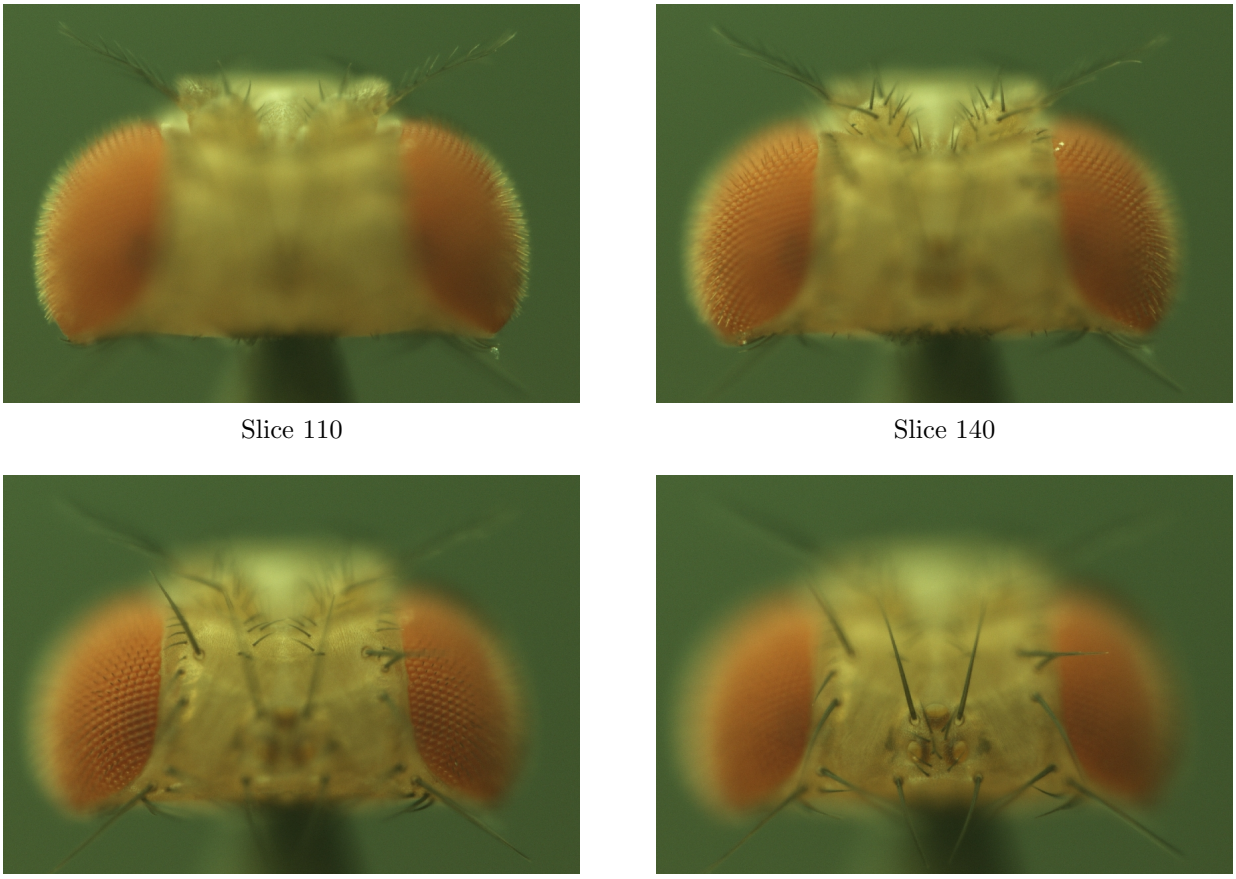



Figure 4.3: Several images in a $z$ stack (300 images) of a drosophila.

underlying geometric shapes (see Fig. 4.3 for examples). These constraints and requirements render most existing three dimensional acquisition methods not applicable.

We have developed a prototype system for estimating three dimensional parts of drosophila based on microscopic image stacks; image stacks with systematic focus changes allow us to estimate the depth through estimation of focus. As the measurements are inherently noisy, we model body parts using thin plat spline models [132], which result in parametric models and can be used for further processing and measurements.

### 4.2.2 A Model of Microscopic Image Stacks

While drosophila are small in size, their external morphology contains very rich features and variations [34]. To illustrate the complexity of drosophila body part forms, Fig. 4.3 shows several

images of one z stack of a drosophila. Clearly these complex phenotypic features make the three dimensional segmentation difficult. The images in Fig. 4.3 also show that we can estimate the depth by estimating the sharpness of a given small region. Here the images are acquired by changing the focus of the microscope systematically where the parts-in-focus show clear details while out-of-focus parts are significantly blurred. By estimating the blur of a small window around each pixel, we can estimate the depth of the pixel, resulting in a range image. Note that also certain features can be occluded by other parts from a particular view and we use multiple stacks when necessary to reconstruct occluded parts.

More formally, given a particular view angle, a drosophila can be modeled by a textured range image. For each pixel $(x, y)$, $z(x, y)$ determines the depth of the model relative to a fixed z coordinate and $c(x, y)$ determines the color of the pixel when the pixel is ideally focused on. To generate an image stack, the z position of the model is varied systematically from $z_1$ to $z_n$, where $n$ is the total number of z positions. Image $I_i$ in the stack is given by

$$I_i = P(z - z_i, c), \tag{4.1}$$

where $P$ is an imaging model of the microscope. Under this formulation, we need to recover both $c(x, y)$ and $z(x, y)$ relative to a fixed but arbitrary origin of $z$. Here the images are automatically registered under the condition that the microscope is static except the movement along the z axis.

As shown in Fig. 4.3, the imaging process $P$ can be approximated by a blurring process (e.g. [105]). The underlying reason for the model is that a typical microscope (as it is the one used for all the experiments) can be modeled by a thin lens model, given by

$$\frac{1}{f} = \frac{1}{d_o} + \frac{1}{d_i}, \tag{4.2}$$

where $f$ is the focus length of the microscope, $d_i$ specifies the image plane relative to the center of the aperture, and $d_o$ is the depth of the object to be ideally focused. When $f$ and $d_i$ are fixed in all the cases here, given $z_i$, then the pixels whose $z(x, y) - z_i$ satisfies Equation (4.2) are in perfect focus; the pixels whose $z(x, y) - z_i$ larger than the ideal $d_o$ or smaller than $d_o$ will become a blurred circle, the radius of the circle or the amount of blur depends on $|z(x, y) - z_i - d_o|$.

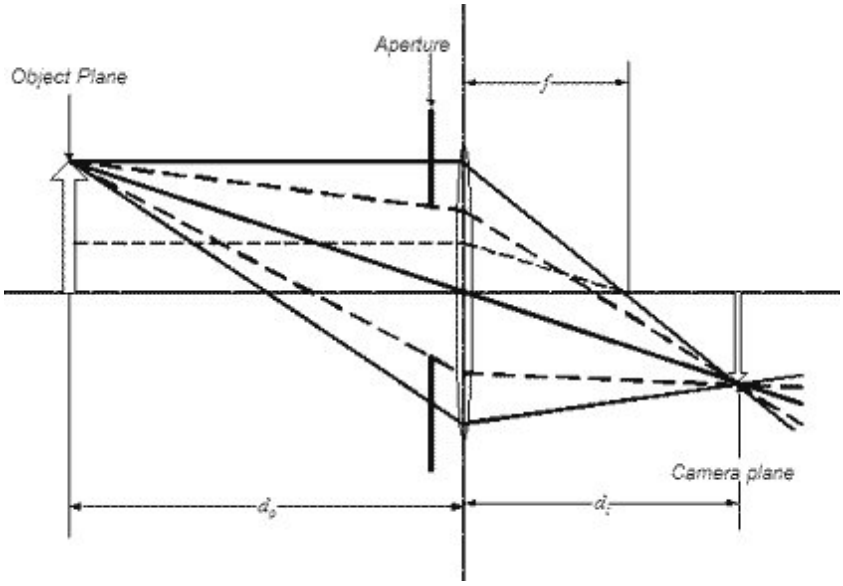



Figure 4.4: Diagram of a thin lens model.

### 4.2.3 Algorithm for Interactive 3D Surface Extraction

Given the model of the stack, we can algorithmically estimate the underlying $z(x, y)$ and $c(x, y)$ from a stack acquired at a particular angle. According to the thin lens model, given $f$ and $d_i$, we can estimate $d_o$ by estimating the amount of blur at a particular pixel. This estimation, however, requires estimation of $f$ and $d_i$ or known $f$ and $d_i$. Here we use a simpler algorithm but does not require known $f$ and $d_i$. As the imaging process $P$ can be approximated by a blurring process, which can be modeled by a Gaussian smoothing with a variance (or the amount of smoothing) that depends on $|z(x, y) - z_i - d_o|$. In a local area with pixel variations, Gaussian smoothing will reduce the variation in the area; so the local variance is the largest when the pixels in the local area are in focus. This leads to an efficient depth estimation algorithm. For each pixel, we simply need to estimate the variance at each pixel and the underlying true depth will be the one with the largest variance.

The accuracy of the above depth algorithm is limited to the step size between two adjacent images with $z_i$ and $z_{i+1}$. For practical purpose, the step size in $z$ has to be as large as possible to avoid large number of images; but a large step size limits the resolution of estimation in $z$. To achieve a step size depth estimation, we fit a quadratic function using variances in a neighborhood

of the largest variance and find the maximum as the estimation of depth. This leads to a sub-pixel depth estimation algorithm but is efficient.

**Parametric Models.** The above algorithm gives a dense point cloud representing the contour of the surface to be reconstructed. However, the sampling is typically noisy due to occlusions and the roughness of the imaged surface. Thus, in order to compare and more reliably estimate shapes, we interpolate a parametric surface through the noisy point cloud for more accurate measurements and estimations. We use a thin-plate-spline (TPS) model for surface segmentation. The TPS interpolator is the function that minimizes the functional

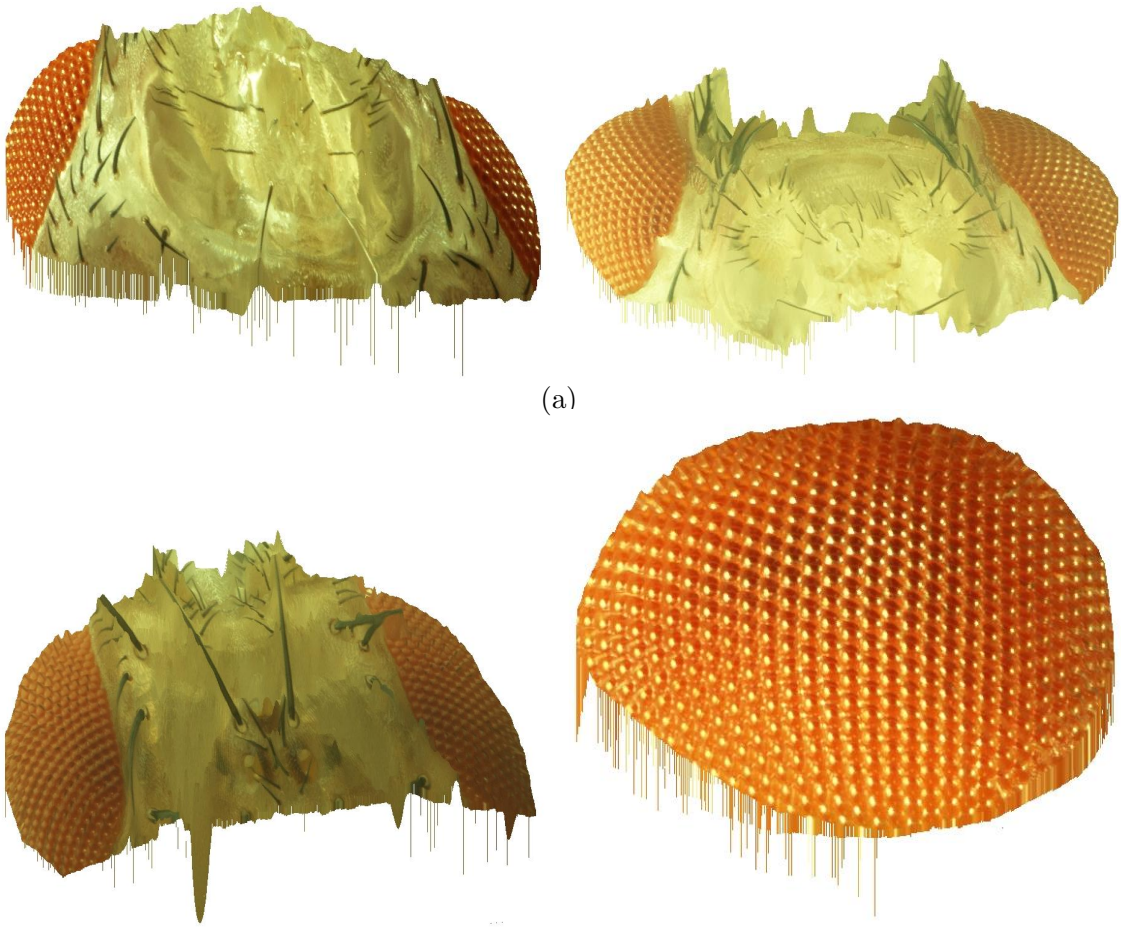
$$E(f) = \frac{1}{N} \sum_{(x,y)} \parallel z(x,y) - f(x,y) \parallel^2 + \lambda J(f), \tag{4.3}$$

on an appropriate reproducing kernel Hilbert space; we refer the reader to [132, 18] for details. Here, $J(f)$ denotes the thin-plate elastic energy, $z(x,y)$ is the estimated depth at pixel $(x,y)$, $f$ is the function to be estimated, $N$ is the total number of points and $\lambda$ is a parameter that controls the smoothness of the model. When $\lambda = 0$, the model will fit the given points tightly. As $\lambda$ increases, the interpolator $f$ will be smoother, but not as faithful to the original data. An optimal $\lambda$-value can be chosen to give a minimal leave-one-out model error, as discussed in [132]. This is the selection criterion for the parameter $\lambda$ used in this paper. An advantage of using the TPS model is that the minimization of the energy can be solved analytically and the problem is reduced to a set of linear equations.

### 4.2.4 Result Visualization

For the experiments, the image stacks were acquired using a Nikon AZ 100, equipped with automatic z-stepping. The entire microscope system can be approximated well by a thin lens model. Depending on the estimated depth range of the drosophila at a particular angle, image stacks were acquired covering the entire depth range.

Figure 4.5 shows several parts of a drosophila from several z stacks from different view angles. The reconstructed range image $z(x,y)$ is texture mapped using the corresponding computed most focused image, which is an estimation of $c(x,y)$. The textures show clearly the depth estimation is accurate for most parts, even though there are noisy depth measurements as most of the pixels are



(a)

(b)

Figure 4.5: Reconstructed different parts of a drosophila from several $z$ stacks of images.

in focus. Note that the results are three dimensional and they can be rendered from different view angles. Here no thin-plate-spline model is applied.

Figures 4.6 and 4.7 show several typical segmented eyes of drosophila from different species, among many examples we have reconstructed. Here the underlying range image $z(x, y)$ is estimated first along with $c(x, y)$ and then we segment the eye part out from the stack. In this paper, an outline was specified manually. The estimation gives a large number of points in each range image. We then estimate the underlying surface using the thin-plate-spline parameter model. In all the cases, the estimated thin-plate-spine model gives a more reliable estimate of the eye component that is more robust and less sensitive to noisy points. It is evident that the reconstructed models characterize well the underlying surfaces and phenotypic features can be extracted for phenotype-genotype relationship study.

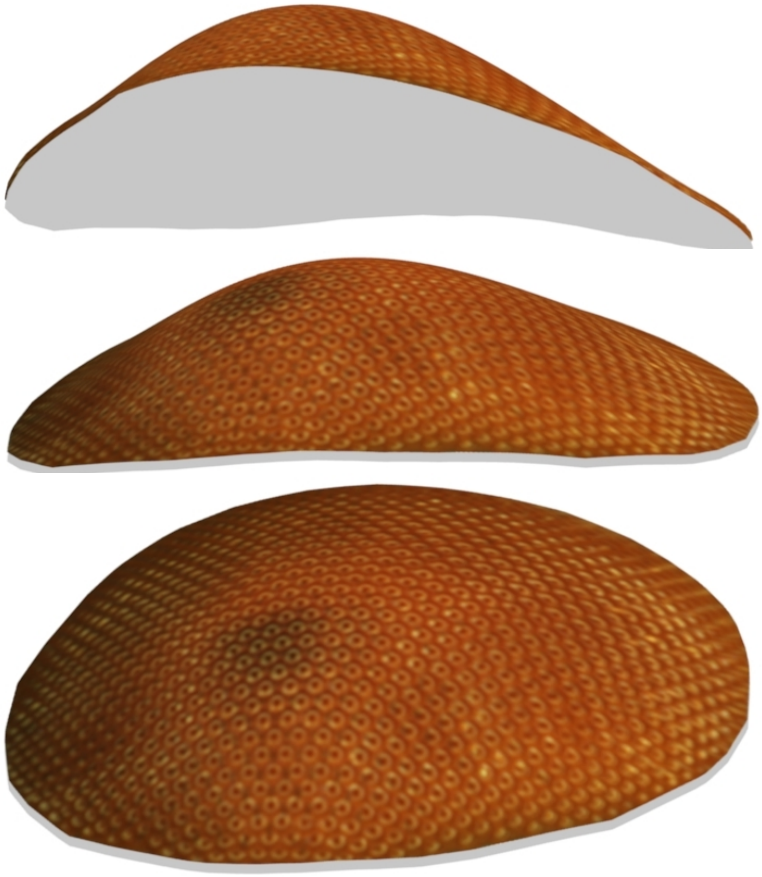


Figure 4.6: A 3D segmentation of a drosophila eye, rendered from different views. Here the segmentation is computed using the thin-plate-spline model of the local surface to remove noise and estimate a parametric surface model that facilitates further processing such as registration and metric reconstruction.

With the estimated parametric surface of a particular body part, phenotypes can be computed using the models and then drosophila with different genetic variations can be compared quantitatively. Clearly the prototype allows systematic study of phenotypic variations with respect to genetic variations and makes large scale studies feasible.

There are several improvements that can be made. Currently each body part to be modeled needs to be segmented out manually. By establishing a common atlas, we can achieve automated segmentation by registering estimated range image to the atlas; this is possible because drosophila are often similar, even though they have complex forms. Another algorithmic improvement is to utilize the estimated amount of blur to estimate the depth (e.g. [105]); an advantage of such a

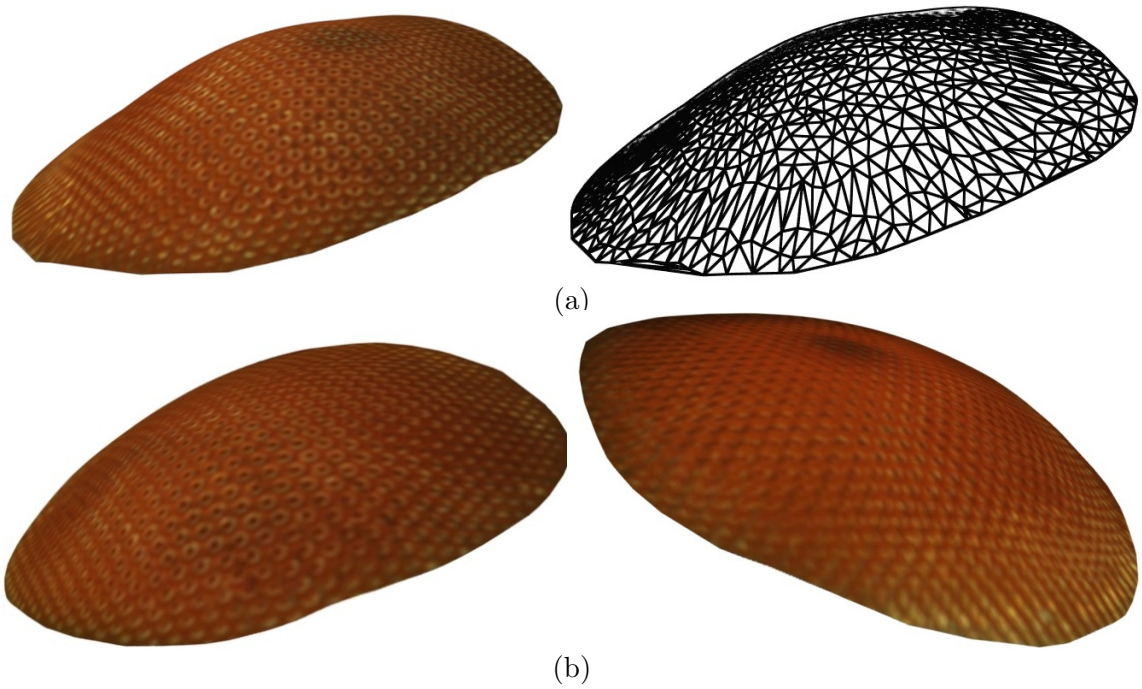


Figure 4.7: More examples of reconstructed eyes: (a) a typical eye with the underlying mesh shown; (b) two more examples of different drosophila.

method is to reduce the number of images needed in a stack for reliable surface estimation. These improvements along with parallel computing, will make high throughput fruit fly surface modeling and large scale phenotyping realizable, which is being investigated.

## 4.3 Interactive Segmentation of HIV Membrane in 3D Space

### 4.3.1 Motivation

The use of cryo-electron microscopes allows the biophysicists to observe nano-scale structures of HIV, such as spikes that the HIV virus uses to bind cells. Thus a possible quantitative analysis of these nano-scale spikes owns great potential in guiding efforts to develop HIV vaccines and treatments. As the first step of quantitative analysis, HIV spike segmentation is inevitable for generating the sub-tomograms containing the HIV spikes.

Based on the fact that the HIV membrane consists of 3D segments that are either ridge-like or edge like in terms of their profiles, I have developed a novel system for segmenting the HIV membrane by circumventing the problem of modeling the HIV membrane. As context information derives from geometric relationship between the context object and the target object, the perfor-

mance of reconstructing the outer surface of the HIV membrane is more important than that of segmenting the exact membrane. Thus my key idea is that, rather than modeling and then localizing the HIV membrane, it is more reasonable and efficient to consider the membrane segmentation as a 3D surface reconstruction problem. Then only the gradient is needed to be involved in an interactive segmentation tool to achieve the outer surface reconstruction.

In our method, a 3D tomogram is considered as a stack of 2D slices along one axis. Thus each outer surface $\hat{S}$ consists of a number of 2D contours $\hat{S} = \{\hat{C}_n | n = 1, 2, ..., N\}$, where $N$ is the number of slices with the contour of the outer surface. The extraction of each connected membrane surface $\hat{S}_k$ is then split into two parts: 1) applying 2D interactive segmentation on the very first slice to extract the initial closed contour $\hat{C}_1$; 2) propagating $\hat{C}_1$ in the 3D space or, in other words, iteratively detecting the contour $\hat{C}_n$ of the current slice based on the gradient information of the current slice and the context constrain from the detected contour of the previous slice ($\hat{C}_{n-1}$). As gradient is a general local appearance feature, our method can be extensively used for other context object segmentation, under the assumption that the context object has a closed contour. In the first part, we use Intelligent Scissor [99], an interactive segmentation algorithm, to extract the desired contour with arbitrary shape in the very first slice. A level set segmentation algorithm is proposed to then segment the contours in nearby slices so as to extend the outer surface in 3D.

Our algorithm differ from existing ones [134, 94] in the manner that the surface is extended from one slice to the next, and in the implementation of the energy functions. With high-resolution data, algorithms working directly in three-dimensional space are really expensive in terms of required computation, and are thus impractical for on-line interactive segmentation, which is more flexible for accurate contour detection of context objects.

Closest to our method is the method of Macke et al. [91], which relies on level set segmentation for the very first slice and then trains a classifier to determine whether a segment belongs to the context object. Therefore, the segmentation of the very first slice can only be captured accurately with available labeled data. Moreover, the extension of the contour in 3D is through the traditional level set method that requires re-initialization in their method. Thus it can suffer from the ad hoc re-initialization [79]. In the following subsections, I will discuss each step of my method in detail.

### 4.3.2 Segmentation of the First 2D Slice

To avoid the problem of context object localization, the Intelligent Scissor was used for generating the closed contour of the HIV membrane outer surface in the very first slice. This slice was manually chosen to be the one with the smallest missing wedge effect (usually the central slice of the HIV tomogram in practice). This is the only slice that requires interactive segmentation. As context objects are large and salient objects in the data, the number of context objects and the respective interactive segmentation operations is quite small. Thus the workload of user input is efficiently controlled. Similar to microvilli membrane segmentation in the previous chapter, we also explore the scale space of the very first image and manually select the scale where the gradient of the HIV outer surface is strong. After applying the Intelligent Scissor, we have the very first closed contour $\hat{C}_1$. The corresponding image plane $\Omega \subset \Re^2$ is thus split into two parts: the foreground region $\Omega_+$ inside the closed contour and the background region $\Omega_-$ outside the closed contour. A sample procedure of 2D segmentation on an HIV outer surface (shown as green contours) is illustrated in Fig. 1.3.

### 4.3.3 Contour Extension in 3D Space

**General Active Contour Model.** To extend the very first 2D closed contour $\hat{C}_1$ toward the 3D outer surface, we need to iteratively detect the contour of the outer surface in the nearby unsegmented slices. In each slice, a proposed evolution will drive an initial boundary toward the desired contour. Here we assume the continuity of the context surface cross nearby slices. Then the detected contour of the precious slice $\hat{C}_{n-1}$ can be utilized to localize the rough location of the desired contour $\hat{C}_n$ in the current slice. This significantly narrows the searching space for contour initialization. Specifically, despite the contour detection of the very first slice, the contour detection in each slice starts from an initial boundary $C_n(0)$ – the boundary of a morphologically dilated mask generated by filling the contour of the nearby segmented slice $\hat{C}_{n-1}$:

$$C_n(0) = \tau(\zeta(\hat{C}_{n-1}) \oplus E_r), \tag{4.4}$$

where $\zeta(x)$ is the morphological filling, $\oplus$ is the morphological dilation, $E_r$ is a ball of radius $r$, and $\tau$ is a morphological operation of boundary extraction on binary image. When the value of $r$ gets

larger, the 3D extension will be less sensitive to continuity, whereas more iterations will be taken to converge.

Concerning the representation of the contour and the implementation of propagating it toward the desired contour, there are basically two ways to model the contour propagation in each slice $\Omega$: parametric active contour and geometric active contour. Both ways achieve segmentation by minimizing appropriate energy functions $E(\cdot)$. Specifically, the parametric active contours (such as Snake [63] and Watersnake [102]) are represented explicitly as dynamic parametric boundaries

$$C(s,t) : [0,1] \times [0,\infty) \rightarrow \Omega. \tag{4.5}$$

Here $s$ is a spatial parameter in $[0,1]$ for the points in the contour and $t$ a temporal variable. The key idea of the propagation is then to evolve the boundary $C(s,t)$ (a number of control or marker points at evolution $t$) from some initialization $C(s,0)$ in the direction of the local negative energy gradient by implementing the following gradient descent equation:

$$\frac{\partial C(s,t)}{\partial t} = -\frac{\partial E(C(s,t))}{\partial C(s,t)}. \tag{4.6}$$

The geometric active contours, on the other hand, are implicitly represented as the zero level set of some function with a higher dimension in an Eulerian framework

$$C = \{x \in \Omega | \phi(\mathbf{x},t) = 0\}. \tag{4.7}$$

Here $\phi(\mathbf{x},t)$ is the level set function parametrized by a spatial variable $\mathbf{x}$ in the image domain $\Omega$ and a temporal variable $t \geq 0$. To avoid unstable evolution and numerical computation error, a signed distance function of the image plane $\Omega$ is usually used as the level set function [28]. The signed distance function determines the distance of a given point $\mathbf{x}$ from the contour in $\Omega$, with the sign determined by whether $x$ is inside the contour. Let's define this function as having positive values at points $\mathbf{x}$ inside the contour ($\Omega_+$). Starting from the center of $\Omega_+$, the signed distance function decreases in value as $x$ approaches the contour of $\Omega$ where this function is equal to zero, and then takes negative values outside of the contour $\Omega_-$. Therefore, it will maintain the level set function neither too flat nor too steep. The key idea of the propagation is then to evolve the level set function $\phi(\mathbf{x},t)$ from some initialization $\phi(\mathbf{x},0)$ in the direction of the local negative energy gradient by implementing the following gradient descent equation:

$$\frac{\partial \phi(\mathbf{x},t)}{\partial t} = -\frac{\partial E(\phi(\mathbf{x},t))}{\partial \phi(\mathbf{x},t)}. \tag{4.8}$$

Correspondingly, the zero level set of the level set function gradually approaches the 2D contour of the outer surface in the current slice. As the implementation of parametric active contour propagation requires a re-griding process to eliminate overlap of control or marker points and lacks a meaningful statistical framework for extensional use, we have proposed a geometric active contour model to solve our problem. For simplicity, we use $h$, $\phi$ and $\phi_0$ instead of the current stop function $h_n(\mathbf{x})$, the current level set function $\phi_n(\mathbf{x},t)$ and the final level set function $\phi_{n-1}(\mathbf{x},t)$ of the previous slice respectively in the rest of this chapter.

**Our Statistical Model.** Let $\phi$ and $g = |\nabla G_\sigma * I|$ be the level set function and the gradient map of a slice $I$ respectively, where $*$ means convolution and $G_\sigma$ is the Gaussian kernel with standard deviation $\sigma$. We want to find an optimized contour represented by the zero level set of function $\phi$ given $g$ and the level set function from the previous (nearby) slice $\phi_0$. According to our statistical framework in Chapter 2, we can formulate our task as maximizing the following posterior probability:

$$\Pr(\phi|g,\phi_0) \propto \Pr(g|\phi,\phi_0) \times \Pr(\phi|\phi_0). \tag{4.9}$$

Here $\phi$ is a representation of the segmentation of the current slice while $g$ and $\phi_0$ can be interpreted as local appearance features and global feature of shape respectively. For simplicity, the gradient map of the current slice $g$ is assumed to be independent of the segmentation of the previous slice $\phi_0$. Thus Eq. 4.9 can be re-written as

$$\Pr(\phi|g,\phi_0) \propto \Pr(g|\phi) \times \Pr(\phi|\phi_0). \tag{4.10}$$

As logarithm is a monotonically increasing function, maximization of Eq. 4.10 is identical to minimization of its negative logarithm, resulting in a general energy function of propagating the contour in 3D space to recover a surface

$$\begin{aligned} E(\phi) &= -\log(\Pr(g|\phi)) - \log(\Pr(\phi|\phi_0)) \\ &= E_{ex} + E_{in}. \end{aligned} \tag{4.11}$$

The first term $E_{ex}$ on the right side is based on the slice information. Thus it is image-dependent and is called an external energy function. This function drives the zero level set of the level set function toward the desired contour. The second term $E_{in}$ is image-independent and is thus called an internal energy function. This function penalizes the deviation of the level set function $\phi$ from its intrinsic properties, such as being a signed distance function and being similar to the contour $\phi_0$ in the previous slice. In what follows, we will specify these two terms in detail.

**The External Energy Function.** Aimed at propagating the initial contour towards the contour of the outer surface in the current slice, we explicitly define an external energy function that is able to move the zero level set of a signed distance function $\phi$ toward this desired contour with strong gradient. To minimize the energy function, we follow the seminal work of geometric active contour [27] and use the stop function based on the gradient

$$h = \frac{1}{1 + |g|^p}, \tag{4.12}$$

where p = 1 or 2. The value of this function decreases when it gets closer to a strong edge. Our external energy function is defined as a gradient-based length term

$$\begin{aligned} E_{ex} &= E_L \\ &= \lambda_L \times \int_\Omega h| \bigtriangledown H(\phi)| d\mathbf{x} \\ &= \lambda_L \times \int_\Omega h\delta(\phi)| \bigtriangledown \phi_n| d\mathbf{x}. \end{aligned} \tag{4.13}$$

Here $H(\cdot)$ is the Heaviside step function

$$H(y) = \begin{cases} 1 & \text{, if } y \geq 0, \\ 0 & \text{, otherwise,} \end{cases} \tag{4.14}$$

and $\delta(\cdot)$ is the corresponding Dirac function

$$\delta(y) = \frac{d}{dy} H(y). \tag{4.15}$$

The term $E_L$ is a modified version of the length term in [29], which was designed to penalize the length of the contour between foreground and background and hence favors smooth curve. By involving the stop function $h$, such term computes the integral along the zero level set of function

$\phi$. Thus it enforces the zero level set of the function $\phi$ to approach contours with strong gradient (where $h$ takes small values) while maintaining the smoothness of the zero level set. By calculation of partial derivative function, the Gateaux derivative (first variation) of the energy function $E_L$ with respect to the level set function $\phi$ is then:

$$\begin{aligned}\frac{\partial E_L}{\partial \phi} &= -\lambda_L \times \delta(\phi) \bigtriangledown \cdot \Big(h \frac{\bigtriangledown \phi}{|\bigtriangledown \phi|}\Big) \\ &= -\lambda_L \times \delta(\phi) \Big[\bigtriangledown h \cdot \frac{\bigtriangledown \phi}{|\bigtriangledown \phi|} + h \cdot K\Big],\end{aligned} \tag{4.16}$$

where

$$K = \bigtriangledown \cdot \Big(\frac{\bigtriangledown \phi}{|\bigtriangledown \phi|}\Big) \tag{4.17}$$

is the curvature.

**The Internal Energy Function.** Even though the level set function is initialized as a signed distance function at the beginning of the evolution for each slice, this property does not hold during the level set function evolution. Such irregularity can develop sharp or flat shape during evolution and thus causes numerical errors, which can finally destroy the stability of the level set function evolution. To overcome the problems due to irregularity, re-initialization is widely used in the literature of the level set methods [27, 92, 28, 29, 128, 91] as a critical and inevitable step. This strategy periodically "reshapes" the level set function $\phi$ to be a signed distance function by enforcing it to satisfy the signed distance property $|\bigtriangledown \phi| = 1$. A standard way to implement this strategy is to solve the following partial derivative equation:

$$\frac{\partial \phi}{\partial t} = \frac{\phi_0}{|\phi_0|}(|\bigtriangledown \phi| - 1), \tag{4.18}$$

where $\phi_0$ is the initial level set function. Ideally, the steady state of this equation is a desired signed distance function that does not only maintain the sign of $\Omega_+$ and $\Omega_-$ but also satisfy the signed distance property. However, re-initialization suffers from the possibility to move the zero level set away from the desired contour. Meanwhile, it is ad hoc to decide when and how to apply re-initialization in practice. Concerning these problem of re-initialization, we follow [79] and involve a regularization term in our internal energy function to implicitly regularize the level set function. With such term, we no longer need to reinitialize the level set function during its evolution.

Moreover, it is important to remain the smoothness (continuity) between contours of nearby slices as our final task in context object segmentation to extract the outer surface in 3D. Thus we need a smoothness term to penalize the difference between the level set functions of adjacent slices. This term helps the propagation of the zero level set contour to favor the contour location of the previous adjacent slice.

Taken the two concerns above into account, our internal energy function is defined as follows

$$E_{in} = E_R + E_S \tag{4.19}$$

such that

$$E_R = \lambda_R \times \int_\Omega \frac{1}{2}(|\bigtriangledown \phi| - 1)^2 d\mathbf{x} \tag{4.20}$$

and

$$E_S = \lambda_S \times \int_\Omega \frac{1}{2} R(|\phi - \phi_0| - \theta)^2 d\mathbf{x}. \tag{4.21}$$

Here $R(y)$ is a half rectified function such that

$$R(y) = \begin{cases} y & \text{, if } y \geq 0, \\ 0 & \text{, otherwise.} \end{cases} \tag{4.22}$$

The first term $E_R$ is designed to register the level set function $\phi_n(x)$ as a signed distance function. As any function $\phi$ that satisfies the signed distance property is the signed distance function plus a constant [4], $E_R$ measures the integral of how close the current level set function is to a signed distance function in image domain. By calculation of partial derivative function, the Gateaux derivative of the energy function $E_R$ with respect to the level set function $\phi_n(x)$ is then:

$$\frac{\partial E_R}{\partial \phi} = -\lambda_R \times (\triangle \phi - K), \tag{4.23}$$

where $\triangle$ is the Laplacian operator.

The second term $E_S$ is a smoothness term that is designed to penalize the deviation of current level set function $\phi_n(x)$ from the final level set function $\phi_{n-1}(x)$ in the previous adjacent slice [91]. Therefore, minimization of this term improves the smoothness between segmentation of nearby

slices by favoring the segmentation which is similar to the one in the previous slice in the slice-by-slice procedure of segmentation. The half rectified function $R(\cdot)$ is employed to allow reasonably small variance of level set functions between adjacent slices. When the variance is smaller than the threshold $\theta$, we will ignore the penalty on the variance. By calculation of partial derivative function, the Gateaux derivative of the energy function $E_S$ with respect to the level set function $\phi_n(x)$ is then:

$$\frac{\partial E_S}{\partial \phi} = -\lambda_S \times R(|\phi - \phi_0| - \theta)\frac{\phi - \phi_0}{|\phi - \phi_0|}. \tag{4.24}$$

**Solution by Gradient Descent.** The way to propagate the zero level set of $\phi$ toward the desired contour in each slice is achieved by iteratively minimizing the total energy function (4.11) in terms of $\phi$. Let $\partial\Omega_+$ be all the points on the initial contour generated by Eq. (4.4), we initialize the level set function of the current slice as a binary step function

$$\phi^0 = \begin{cases} c_0 & \text{, if } \mathbf{x} \in \Omega_+, \\ -c_0 & \text{, otherwise,} \end{cases} \tag{4.25}$$

where $c_0$ is a constant scalar. For the iteration other than the first one, based on Eq. 4.8, we have

$$\frac{\phi^{n+1} - \phi^n}{\triangle t} = -\frac{\partial E(\phi)}{\partial \phi}, \tag{4.26}$$

where $\triangle t$ is the time step. Then we have the following equation that updates $\phi$ and thus propagates its zero level set toward the desired contour:

$$\begin{aligned} \phi^{n+1} &= \phi^n - \triangle t \frac{\partial E(\phi)}{\partial \phi} \\ &= \phi^n - \triangle t(\frac{\partial E_{ex}(\phi)}{\partial \phi} + \frac{\partial E_{in}(\phi)}{\partial \phi}) \\ &= \phi^n - \triangle t(\frac{\partial E_L(\phi)}{\partial \phi} + \frac{\partial E_R(\phi)}{\partial \phi} + \frac{\partial E_S(\phi)}{\partial \phi}). \end{aligned} \tag{4.27}$$

**Post processing.** The result of the previous step is a 2D binary map (zero level set) indicating the pixels of the current slice that potentially belong to the outer surface of the selected membrane. Without knowing the first and the last slice containing the contour of the outer surface beforehand, this step is carried out across all the slices. To remove the false positives in the 3D binary map, our aim in the post processing is to identify the actual outer surface voxels based on semi-global

gradient analysis. As a feature shared by context object, the size of the membrane outer surface makes it distinctive from the cluttered background. Thus the gradient in terms of a semi-global region makes the outer surface more salient. After smoothing the gradient values of the potential surface voxels in terms of its local surface region, a threshold $t_g$ is then introduced to control the voxels that are considered as the ones on the actual membrane outer surface. Again, the single thresholding could be replaced the hysteresis thresholding strategy to improve the connectedness of each membrane outer surface. Finally, a morphological opening operation is applied to smooth the outer surface in that the existence of spikes arrayed on the surface may destroy the smoothness of the outer surface.

## 4.4 Experiments

### 4.4.1 Visualization of Evolution in 2D Slices

For the experiments, the tomogram was acquired using a cryo-electron microscope. The $z$ axis is the direction parallel to the electron beam, along which the missing wedge effect gives rise to a loss of the resolution. Hence our tomogram is considered as a stack of 2D slices along the $z$ axis.

Fig. 4.8 shows the evolution of the level set functions on several typical HIV membranes in the 86'th slice, with their respective zero level sets shown as red curves in Fig. 4.9. Here the 87'th slice is manually chosen as the very first slice as the missing wedge effect on this slice is quite small. The interactive Intelligent Scissor was applied on slice 87 at first. After the operation of Eq. 4.4 on the segmentation results of slice 87, we have the initial contours $c^0$ for the next slice (slice 86). Due to the assumption that the changes of slices throughout the membranes are quite small, the parameter $r$ in Eq. 4.4 does not need to be very large. In practical, we find $r = 5$ is large enough to tolerate the changes among slices in our tomogram. Analytically, the larger the change is among slices, the larger the value of $r$ should be, along with more iterations needed to reach a stable energy of the evolution.

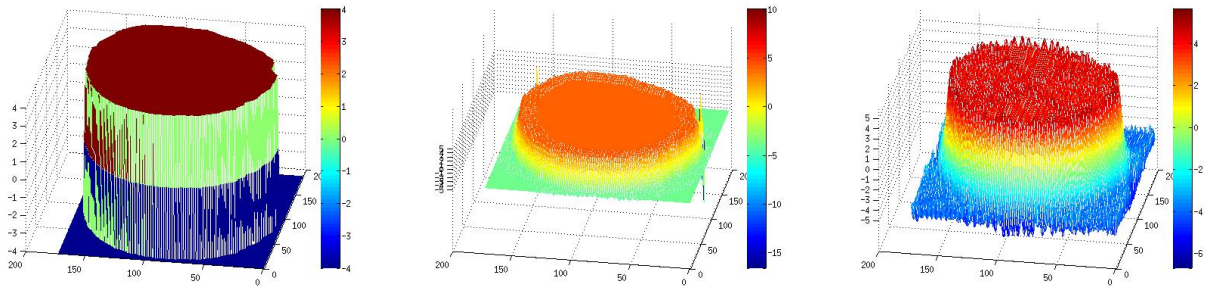
The first column of Fig. 4.9 illustrates three sample initial contours $\partial\Omega_+$ in slice 86, with their corresponding initial level set functions $\phi^0$ shown in the first column of Fig. 4.8 by applying the operation in Eq. 4.25 on $c^0$. The membrane in the first row is the simplest case in which the textures inside the membrane are close to homogeneous. The profile of the membrane is ridge-like. In the second row, the membrane is also ridge-like. But the texture inside the membrane is no

longer homogeneous because of the existence of some other tissues. For the third case (the third row), the upper-right part of the membrane is edge-like rather than ridge-like due to some inner tissues attached to the membrane. In all three cases, our zero level set succeeded in reaching the desired contour after the revolution. Thus our model based on the gradient of the outer surface of the membrane and the prior from the previous slice gives a reliable segmentation that is more robust and much less sensitive to the noisy inside of the membrane. To visualize the function of the length term based on gradient, we plot the external energy in the first column of Fig. 4.10 (the red solid line). It is obvious that our external energy gradually enforces the zero level set to approach the ideal contour by minimizing the integral on Eq. 4.12, or in other words maximizing the integral on the gradient map while encourage the smoothness of the curve.

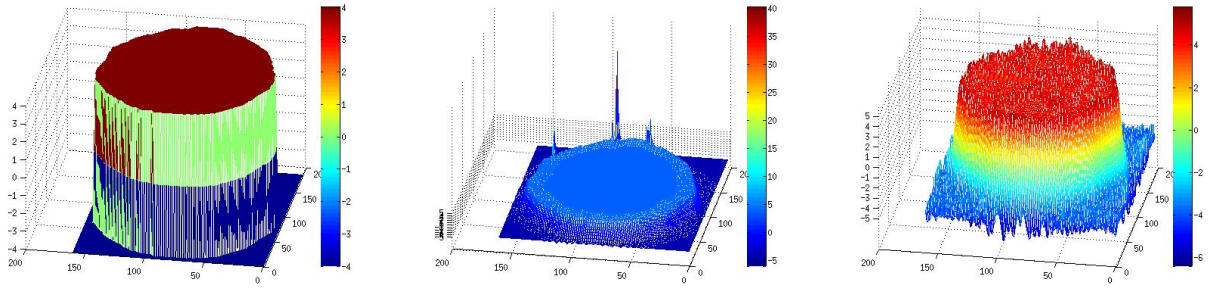
As we discussed in section 4.3.3, the level set function may produce either sharp or flat shape during evolution because of irregularity. Fig. 4.8 (b) shows an exemplar intermediate state where our level set function is too flat, whereas Fig. 4.8 (e) shows an exemplar intermediate state where there is a sharp jump in our level set function. Because of our regularization term in Eq. 4.23, the level set functions are both intrinsically and automatically regularized and thus the final states (the last column of Fig. 4.8) are all neither too flat nor too steep. To visualize the function of the regularization, we also plot the regularization energy in the first column of Fig. 4.10 (the blue dash-dot line). It is clear that our regularization term keeps adjusting the shape of our level set function by steadily minimizing the difference of our level set function from being a signed distance function. By automatically minimizing the total energy consisting of these three energy terms simultaneously, shown as the green solid lines in the second column of Fig. 4.10, the evolution allows us to extract desired closed object contours in 2D.

### 4.4.2 Visualization of Contour Extension in 3D Space

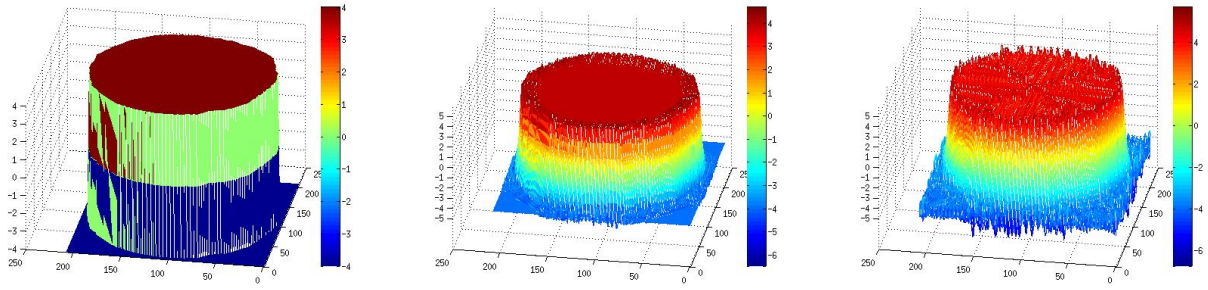
Figure 4.11 shows a number of sampled slices from the final 3D segmentation of the given tomogram. Again, green curves delineate the contours of the outer membrane surface in such slice. By involving our smoothness term (4.24), the segmentation is shown to still reach the desired contours after propagating the contour in slice 87 (the very first slice) for more than 40 slices, regardless of the noisy and the low contrast (i.e.: the segmentation of slice 40 and 120 in Fig. 4.11). To visualize the function of the our smoothness term, we also plot the smoothness energy in the first column of Fig. 4.10 (the magenta dashed line). It is clear that our smoothness term steadily



(a) (b) (c)



(d) (e) (f)



(g) (h) (i)

Figure 4.8: Sample evolution of level set function on 3 different membranes. Each row illustrates the evolution for one membrane. The initial level set function, an intermediate level set function and the final level set function are shown from left to right.

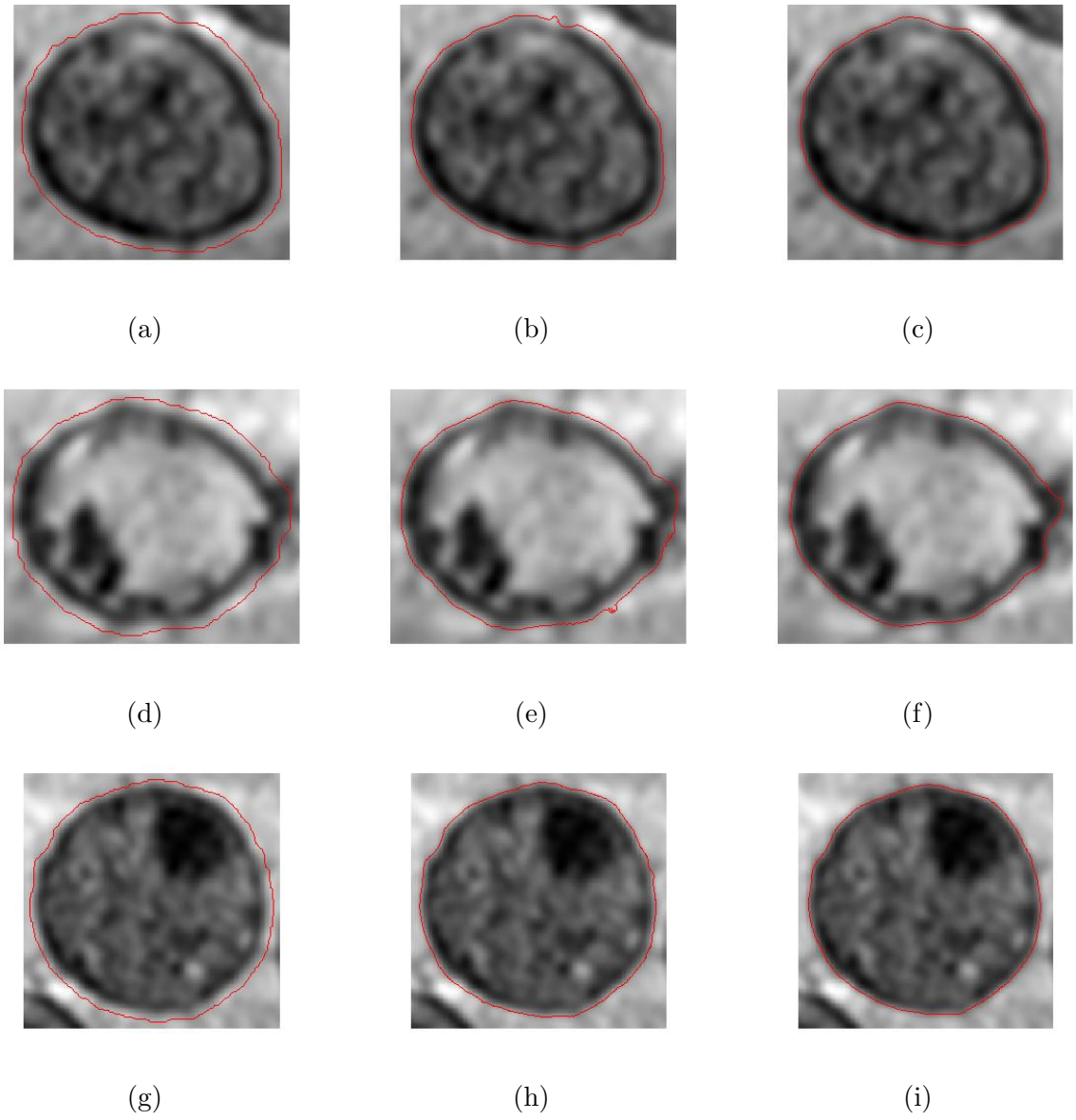



Figure 4.9: The respective zero level sets of the level set functions in Fig. 4.8 (red curves). For visualization, we only show the local window of the current slice that contains one member. Again, each row illustrates the evolution for this membrane. The zero level sets of the initial level set functions, the intermediate level set functions and the final level set functions are shown from left to right.

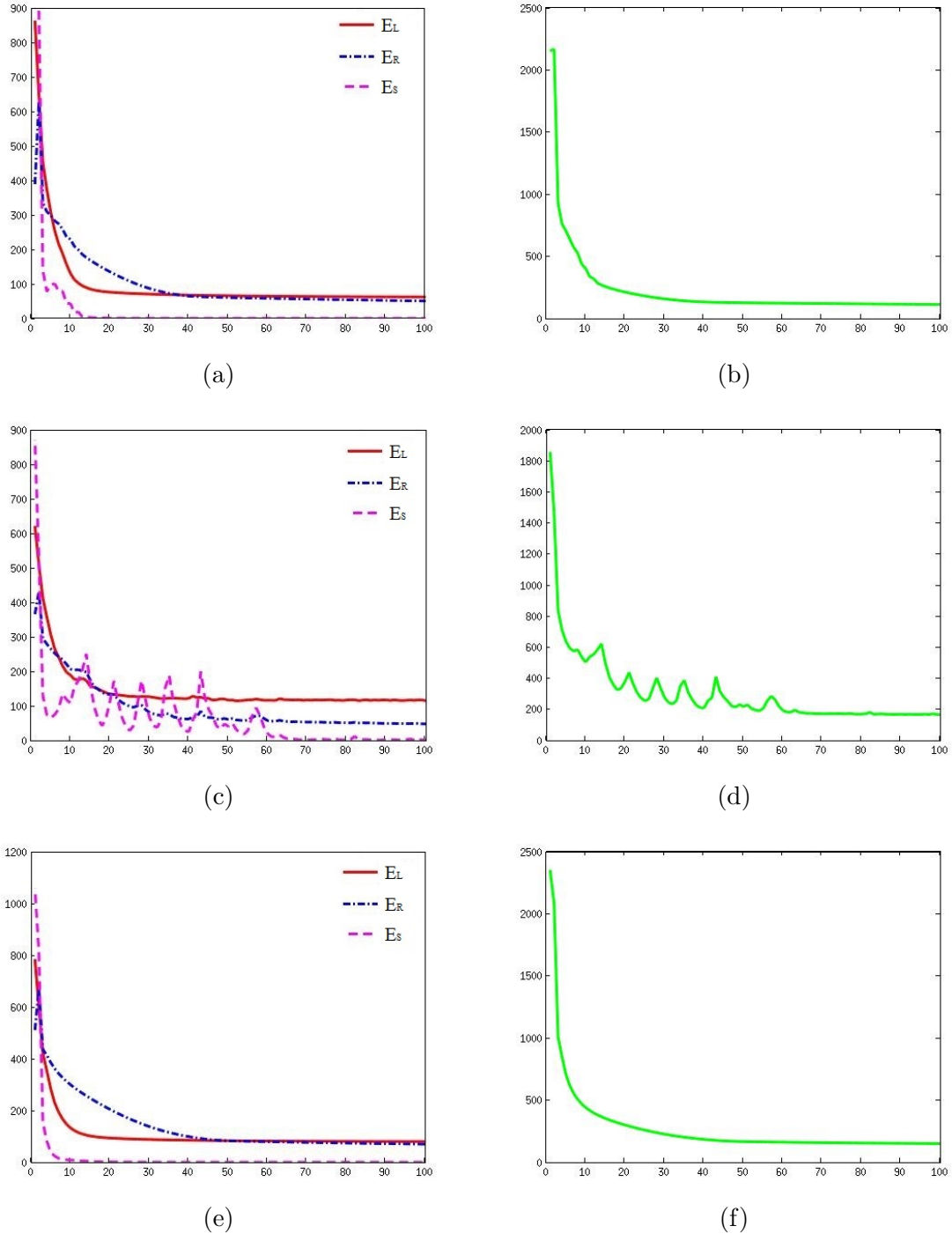



Figure 4.10: The respective energies of the level set functions in Fig. 4.8. Again, each row illustrates the evolution for one membrane. The first column shows the plots of $E_L$, $E_R$ and $E_S$, whereas the second column shows the plots of the total energy.

minimizes the distance between our level set function and the level set function deriving from the segmentation of the previous slice.

While Fig. 4.11 shows the accuracy of our method on extracting the outer surface of the membrane, Fig. 4.12 visualizes the whole 3D point cloud of the outer membrane surfaces from two view angles. As each color labels one connected component in 3D, it shows clearly the contours in 2D are connected throughout the tomogram to represent the voxels belonging to the same outer surface as a whole because of our smoothness term. Thus the spatial relationships among different membranes are well preserved for the connected component analysis. Another valuable observation in 3D space is a 3D view of the missing wedge effect. For example, it is easy to observe a hole inside the indigo membrane outer surface in the second view of Fig. 4.12. This is because of missing information on two sides of this membrane along the $z$ axis.

With the segmented outer surface of each membrane, the normal of the surface on each voxel can be computed easily and therefore provides potential spatial context cues of spikes arrayed on the outer surface. Clearly the segmentation allows systematic study of nano-scale membrane with respect to variations of inside texture and makes large scale studies feasible.

## 4.5 Summary

In the first part of this chapter, a prototype system has been proposed to reconstruct the outer surface of salient object in light microscopic images. Such system facilitates systematic studies of the relationship between the phenotype and genotype using the drosophila as the model organism. Due to short development cycle and easy genetic manipulations, the drosophila is an ideal model organism that is widely used to model certain human diseases; additionally, understanding of the phenotype and genotype map can further our understanding of evolution biology and better model fundamental aspects of biology as many genetic traits are preserved in the drosophila. To derive phenotypes, we estimated a model (consisting of a range image and texture map) from a particular view angle by estimating the most focused pixels along the stack. The algorithm is derived based on a thin lens model. The estimated depth points are then fitted using a thin-plate-spline parameter model that is used to compute reliable and stable outer surface of salient object in light microscopic images.

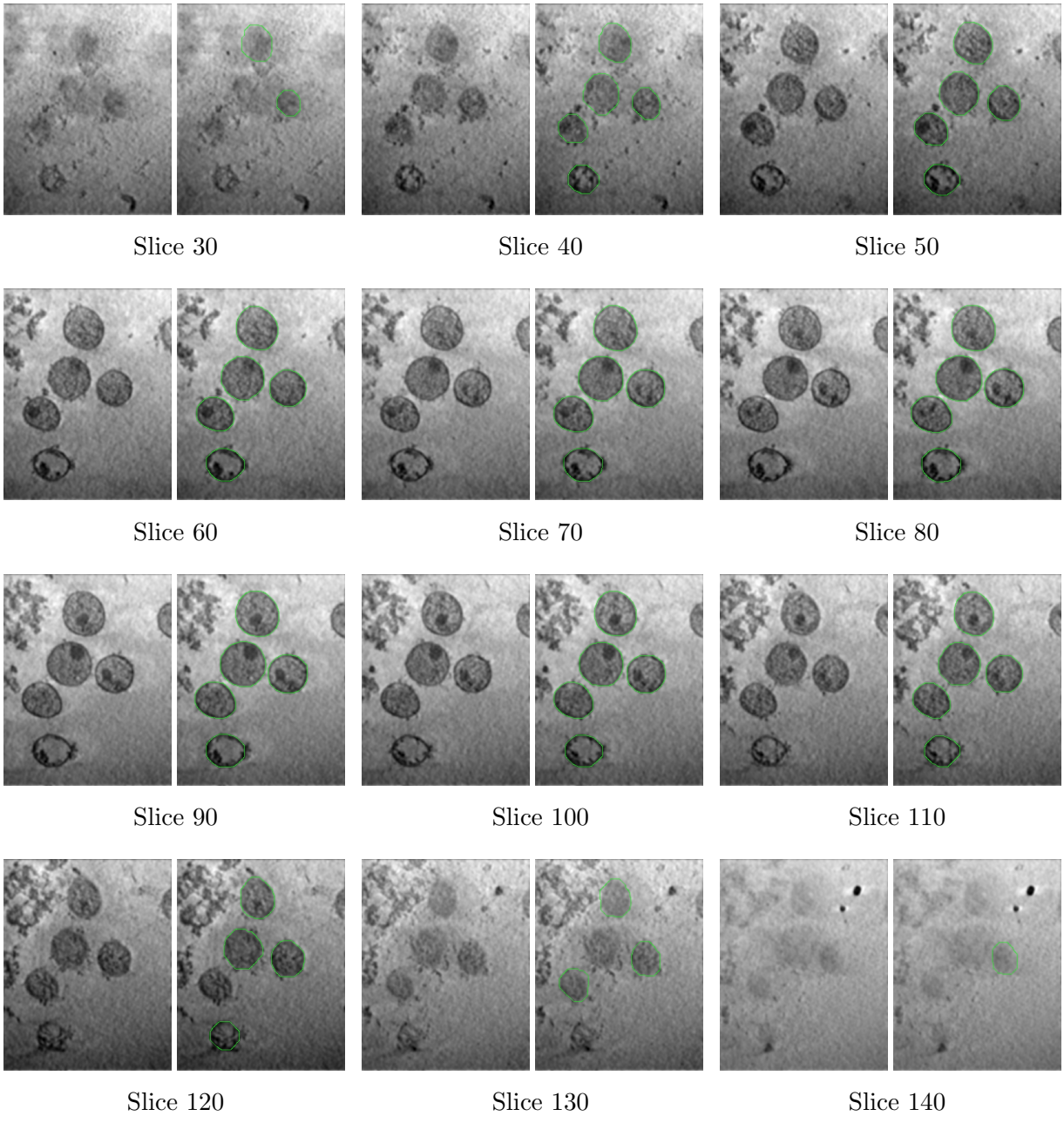
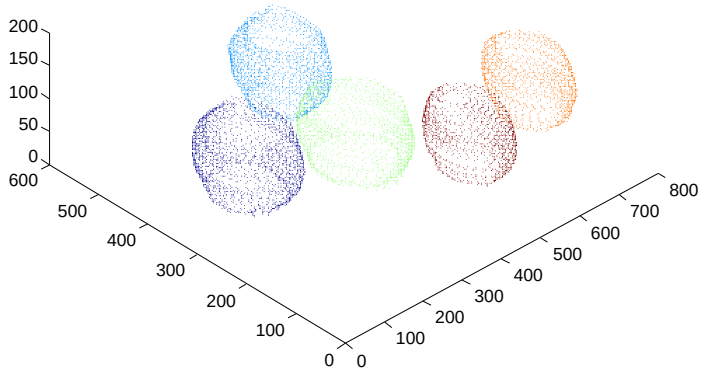
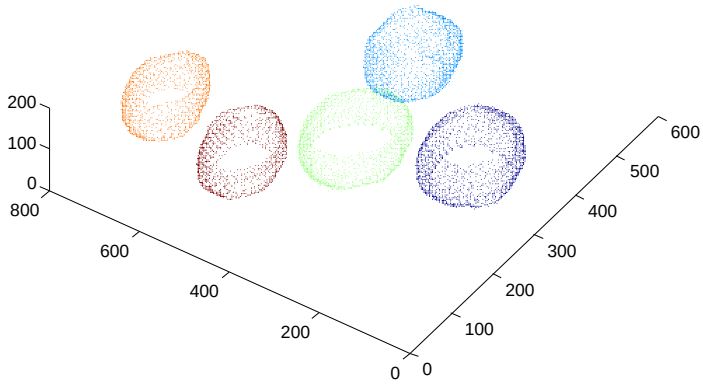



Figure 4.11: Sample 2D slices illustrating the automatic extension of the contours in slice 87 (the very first slice) throughout the entire 3D space. For each sample slice, there is a pair. The original slice is shown on the left and the one with the extracted contours (green curves) is shown on the right.




View 1




View 2

Figure 4.12: Illustration of membrane outer surface reconstruction from two views in 3D. Each color is associated with one membrane outer surface.

In the second part of this chapter, a new HIV membrane segmentation algorithm is proposed to be robust to cluttered background both inside and outside the membranes. The shape of the level set function is maintained by a regularization term, whereas the segmentation smoothness across the tomogram is favored by a smoothness term. By using low-level appearance feature (gradient), our algorithm is also general enough to extract membranes with different profile shape. Our experiments have shown that our segmentation result does not only capture the accurate contours in 2D but also maintain small changes of the 2D contours throughout the tomogram. With the reliable extraction of membranes, in the next chapter, we will discuss the possibility of

designing and utilizing the context cues from membrane segmentation for a even more challenging problem – spike segmentation.

# CHAPTER 5

# 3D CONTEXT-SENSITIVE SPIKE SEGMENTATION

## 5.1 Introduction

In Chapter 3, a salient context object segmentation stage is applied by running various filters over the scale-space for microvillus membrane segmentation. By taking into account the existence of the contextual cues provided by the membranes, the searching space for the target spikes in the original resolution is considerably reduced. In the previous chapter, we extended the ability of our first stage to deal with context object in more general cases. It thus creates the possibility for us to compute contextual cues over extended neighborhoods for a small amount of all the voxels in the original resolution. As our contextual cues are distinguishable to noise, they allow us to apply segmentation strategies such as thresholding to produce per-voxel semantic segmentation and achieve dramatic improvement in detectability.

In this chapter, we focus on the task of spike segmentation based on contextual cues. This task is illustrated in Fig. 5.1, where two sample spikes in a 2D slice are marked by yellow windows. There are three primary sub-problems in this chapter: 1) what the possible context cues are; 2) how to generate them efficiently; 3) how to design a model that combines all these cues. In what follows, we describe the details for calculating the conditional probability of each voxel belonging to a spike in terms of different type of cues, following the same notation as in Chapter 2.

## 5.2 Appearance Cues

As spike heads are somewhat darker than the local background in certain scale space, they appear as local minima in the 3D tomogram. Thus we process tomogram $I$ by smoothing it with isotropic Gaussian filter $G$ of variance $\sigma'$, $H = I * G_{\sigma'}$, and then generate the appearance model:

$$\Pr(o_i{}' = 1 | f_i^{A'}) = \begin{cases} \psi(H_i) & \text{, if } i = \arg\max_{j \in \mathcal{N}_i} H_j, \\ 0 & \text{, otherwise,} \end{cases}$$

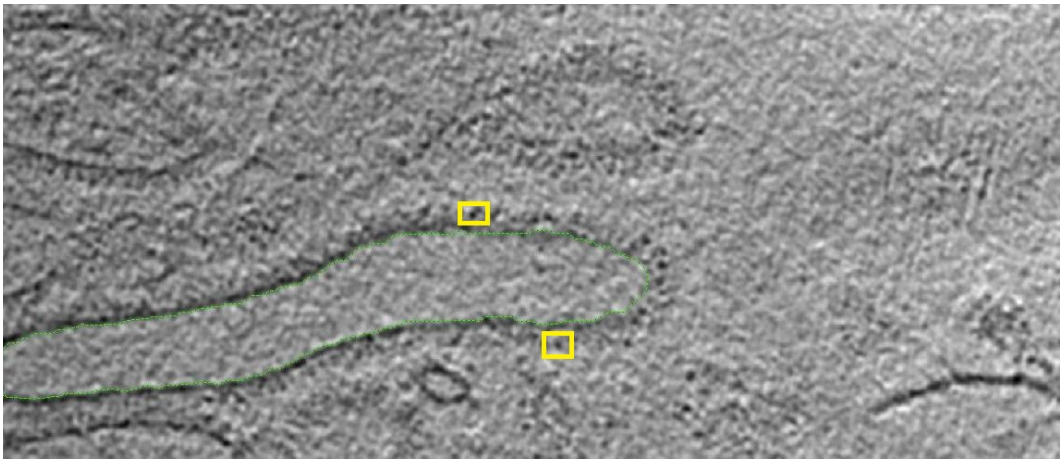


Figure 5.1: Illustration of our task in this chapter, spike segmentation on the exemplar slice in Fig 3.1. The green curve indicates a membrane and yellow windows show two sample spikes arrayed on this membrane.

such that

$$\psi(H_i) = \frac{\max(H) - H_i}{\max(H) - \min(H)}. \tag{5.1}$$

Here $\mathcal{N}_i$ is the $i'th$ voxel with its 26 neighbor voxels in 3D tomogram. An exemplar slice with local minima marked as red crosses can be seen in Fig. 5.2(b).

## 5.3 Context Cues

### 5.3.1 Scale Context

One potential problem of appearance feature in 5.1 is due to the fact that membrane voxels and noise voxels in the background may also appear as local minima. Moreover, the membrane voxels are as dark as or even darker than the spike voxels. Thus their inclusion can deteriorate the spike segmentation performance.

To reduce these false-positive local minima, we rely on the scale context cues to detect spikes in more likely spatial locations and scales. For a microvilli tomogram, spikes are always perpendicular to the surface of the arrayed membrane and the ratio between the length of the spike and the thickness of the membrane is often known (e.g., approximately 20 : 3 for microvilli). This scale context may facilitate faint target segmentation in that it significantly reduces the need of multi-

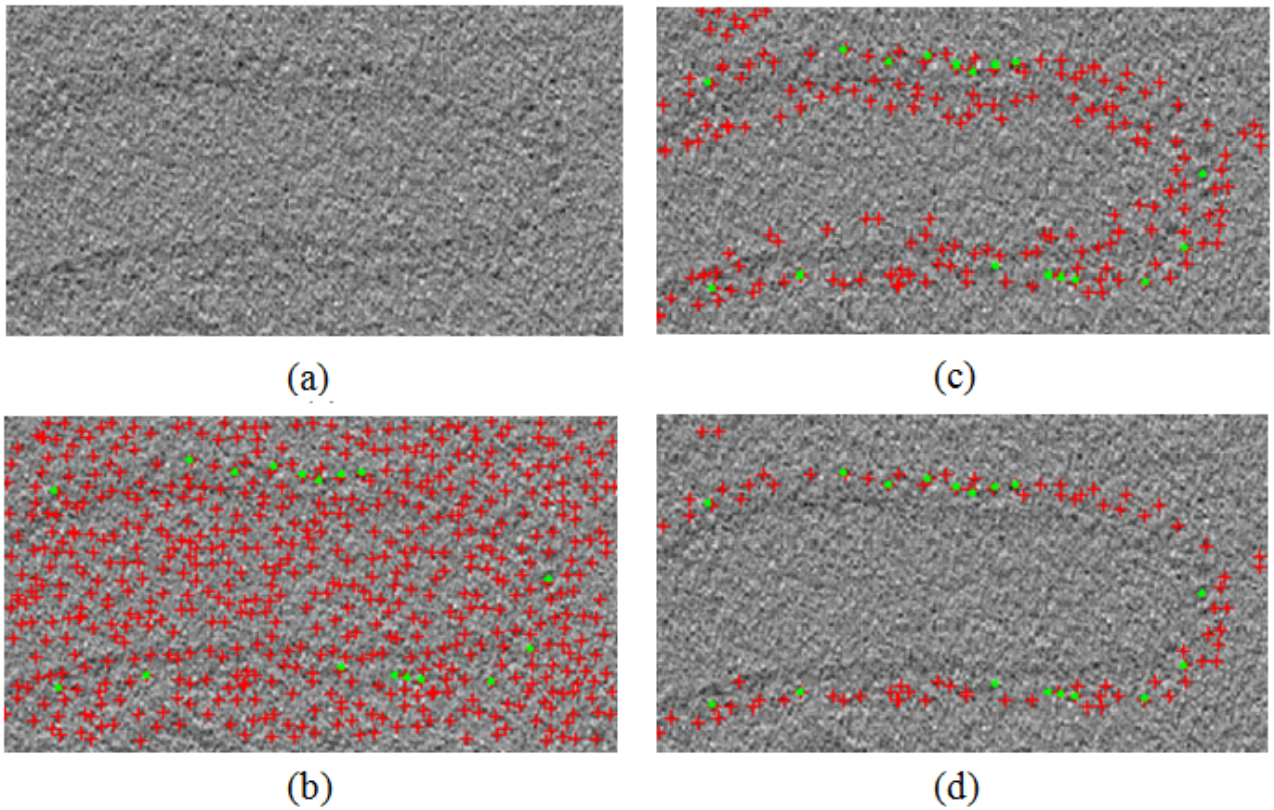



Figure 5.2: Illustration of potential spike head pools after applying each feature sequentially on a single slice. Here red crosses indicate voxels in the potential spike head pools, whereas green dots mark the ground truth spike heads annotated by the expert. From top to bottom, the figures are (a) the original image $I$, (b) the pool concerning only $f^{A'}$, (c) the pool concerning $f^{A'}$ and $f^{sc}$, (d) the pool concerning $f^{A'}$, $f^{sc}$ and $f^{sp}$.

scale search [47] and hence focuses on the appropriate scale. Let $t$ and $h$ be the thickness of the membrane and the maximum possible length of a spike respectively. Given the 3D binary mask of all membranes $M = \{M_k\}$ from the previous chapter, we need to form a number of zones to exclude the local minima due to not only membranes but also background noise that is far from the membrane in contrast to spike heads. These exclusion zones are computed by morphologically dilating the 3d membrane mask $M$ by two balls: $E_t$ of radius $t$ and $E_h$ of radius $h$. The difference between these two dilated masks, as shown in Fig. 5.2(c), defines the scale context feature for spike:

$$f_i^{C_{sc}} = M \oplus E_h - M \oplus E_t, \tag{5.2}$$

where $\oplus$ denotes the 3D morphological dilation. Correspondingly, the likelihood of scale context feature is as follows:

$$\Pr(f_i^{C_{sc}} | f_i^{C_{se}}, o_i' = 1) = \begin{cases} f_i^{C_{se}}, & \text{if } f_i^{C_{sc}} = 1, \\ 1 - f_i^{C_{se}}, & \text{otherwise,} \end{cases} \tag{5.3}$$

where $f_i^{C_{se}} = 1$ if the root of the spike that contains voxel $i$ is labeled as 1 in membrane segmentation result $M$, whereas $f_i^{C_{se}} = 0$ if the respective root is labeled as 0 in $M$. In our scale context model, we formulate the fact that the scale context cue must be satisfied ($f_i^{C_{sc}} = 1$) if the labels of the target (the spike head) and the context (the respective spike root on the membrane) are both given. On the other hand, if the scale context cue is not satisfied ($f_i^{C_{sc}} = 0$) given the target label, the corresponding context label must be mis-labeled as 0 in $M$.

### 5.3.2 Spatial Context

Another problem is that appearance features may lead to unnecessarily exploring the inside of membranes, which ignores the prior knowledge that no spike exists inside and thus causes not only inefficiency but also detection errors.

To overcome the problem by inside search, we also design a spatial context feature, describing the spatial relationship between the context object and the faint target. As spikes are known to grow perpendicularly towards the outside of the arrayed membrane, the shape of which is convex in general, we make an intuitive assumption that the centroid index $c_k$ of every membrane mask $M_k$ is inside the membrane. In that sense, it is most likely that the root of each spike (where the spike is attached on the membrane) should be closer to the membrane centroid than the spike head. An

example is shown in Figure 5.2(c). Let $d(.,.)$ be the Euclidean distance between two voxels given their indexes. We can compute the spatial context feature as:

$$f_i^{C_{sp}} = \frac{d(c_k, i)}{d(c_k, i'_M)}, \tag{5.4}$$

where $i$ is the index of the potential spike head and $i'_M$ is the index of the corresponding spike root on membrane segmentation $M$. As we assume the local membrane as a plane, localizing the spike root $i'_M$ is approximated by finding the membrane voxel in $M$ that is closest to the spike head. As $f_i^{C_{sp}}$ is larger than 1 if the voxel $i$ is outside the membrane, the likelihood of spatial context feature is as follows:

$$\Pr(f_i^{C_{sp}} | f_i^{C_{se}}, o_i{}' = 1) = \begin{cases} f_i^{C_{se}}, & \text{if } f_i^{C_{sp}} > 1, \\ 1 - f_i^{C_{se}}, & \text{otherwise.} \end{cases} \tag{5.5}$$

The definition of $f_i^{C_{se}}$ is the same as in Eq. (5.3). Similarly, here we model the spatial context cue that a spike root must be closer to the center of the arrayed membrane than its respective spike head is, indicating the outside of the membrane. Vise verse, if the spatial context cue is not satisfied ($f_i^{C_{sp}} <= 1$) given the target label, the corresponding context label must be mis-labeled as 0 in $M$.

### 5.3.3 Semantic Context

Clearly, both our scale context model and spatial context model depend on the semantic context cue $f_i^{C_{se}}$. This is straightforward in that it is impossible to obtain any reliable context cues given context that is erroneously identified. In microvilli tomogram for example, due to the intrinsic property of nano-scale imaging, parts of the membranes $M$ are heavily blurred or even missed because of noise and missing wedge effect. The spikes, on the other hand, may maintain somewhat of the contour that is similar to the arrayed membrane and are thus marked as membrane voxels in $M$, especially when they are dense enough. In such region, our scale context feature and spatial context feature both produce false negatives for spike segmentation.

In order to reduce the effect of nano-scale imaging on spike segmentation, one way is to take the membrane likelihood channel $g_m$ into account, as a semantic context cue that represents the co-existence of the context object and the target based on the membrane likelihood. Specifically, a reliable membrane mask $M'$ could be defined as

$$M' = H(M \times g_m - \alpha), \tag{5.6}$$

where $\alpha$ is a threshold that controls the confidence of the membrane segmentation and $H(.)$ is the nonlinear heaviside step function in (3.1). Correspondingly, the scale context feature (5.2) is replaced by

$$f_i^{C_{sc}} = M \oplus E_h - M' \oplus E_t, \tag{5.7}$$

and the spatial context feature (5.4) is remodeled as

$$f_i^{C_{sp}} = \frac{d(c_k, i)}{d(c_k, i'_{M'})}, \tag{5.8}$$

where $i'_M$ is the index of the corresponding spike root on reliable membrane segmentation $M'$. However, the disadvantage of this model is that the contribution of the semantic context cue on semantic segmentation is somewhat not quite straightforward.

Instead of intrinsically modeling the semantic context in the context object segmentation, we explicitly model the semantic context cue as the coefficient in a hybrid model, determining the relative contribution of appearance features and context features in semantic segmentation. Specifically, we directly model the likelihood of the semantic context feature as follows:

$$\Pr(f_i^{C_{se}} | o'_i = 1) = \begin{cases} \lambda, & \text{if } f_i^{C_{se}} = 1, \\ 1 - \lambda, & \text{if } f_i^{C_{se}} = 0. \end{cases} \tag{5.9}$$

If we assume the scale context feature $f_i^{C_{sc}}$ and the spatial context feature $f_i^{C_{sp}}$ are conditionally independent of each other given the semantic context feature and the target label, the spike likelihood channel, Eq. 2.6, is further modeled as:

$$\begin{aligned} &\Pr(o'_i = 1 | f_i^{A'}, f_i^{C_{sc}}, f_i^{C_{sp}}, f_i^{C_{se}}) \\ &\propto \Pr(o'_i = 1 | f_i^{A'}) \times \Pr(f_i^{C_{se}} | o'_i = 1) \times \Pr(f_i^{C_{sc}} | f_i^{C_{se}}, o'_i = 1) \times \Pr(f_i^{C_{sp}} | f_i^{C_{sc}}, f_i^{C_{se}}, o'_i = 1) \\ &= \lambda \Psi^C + (1 - \lambda) \Psi^A, \end{aligned}$$

such that

$$\Psi^C = \Pr(o'_i = 1|f_i^{A'}) \times \Pr(f_i^{C_{sc}}|f_i^{C_{se}} = 1, o'_i = 1) \times \Pr(f_i^{C_{sp}}|f_i^{C_{se}} = 1, o'_i = 1),$$
$$\Psi^A = \Pr(o'_i = 1|f_i^{A'}). \tag{5.10}$$

(see Appendix B for a detailed proof). In this model, the semantic context cue is explicitly formulated by $\lambda$, which we call *sensitivity coefficient*, describing how sensitive the target segmentation is to the context cues. When $\lambda$ is equal to 0, meaning no contribution from context for segmentation, this model is identical to the classic object-centered semantic segmentation. Thus the classic object-centered semantic segmentation is a special case of our model. When $\lambda$ increases, more contributions from the context cues are taken into account in semantic segmentation and the segmentation is thus more sensitive to the context. When $\lambda$ is equal to 1, our model is close to the context integration models [122, 123, 125].

## 5.4 Experiment on Microvillus Tomogram

In this experiment, we evaluated our semantic segmentation method on a 3D tomogram of microvilli. The performance of our method was assessed on the task of spike head detection.

In this section, we first describe the dataset and our evaluation methodology. We then use our dataset to evaluate the accuracy of our method in terms of the number of spikes that are correctly detected. As we have known so far, there is no other method working on segmenting microvillus spikes algorithmically. Hence we use the method based on only local appearance as a baseline and compare it with our method.

### 5.4.1 Dataset and Evaluation Methodology

The dataset was acquired from the microvilli of insect flight muscle. We have a volume with a size of $600 \times 1400 \times 432$, or nearly 363 million voxels. An example slice is shown in the top image of Fig. 3.1. As mentioned previously, there is a large number of faint and small spikes (named spikes for microvillus) in the tomogram, which is quite noisy, the annotation is extremely time-consuming. Thus it is difficult to annotate all the spikes in such a tomogram. To reduce the labeling cost, a microvillus expert partially annotated 89 spike heads on one target membrane in our dataset, leaving most of the spikes unannotated. To avoid false positives caused by unannotated spikes, the performance of our method was assessed on two slices where most of the spikes arrayed on the target membrane (27 in total) were annotated.

**Data Annotation.** Spike labeling is an ill-posed problem. There are ways like 3D binary mask for spike roots, spike centers, the entire spike profiles, or spike heads. Even though from Section 5.3.2 we can have scores for all the membrane voxels indicating its possibility of being a spike root, the validation based on the roots of spikes is not straightforward. As spikes usually only stand out from the background because their headers appear dark enough, their roots are usually indistinguishable from the background. Thus benchmarks for spike roots are unreliable. For the same reason, benchmarks for spike centers are not preferred for validation of segmentation algorithm as well. The last representation, marking the entire spike profile, is ideal for classic segmentation [93] as it is straightforward to generate the standard validation measure – the precision-recall curve. Unfortunately, due to the low SNR, extremely heavy workload is required for experts to repeatedly refine parameters so as to manually mark every spike voxel. And it is also impossible for experts to accurately mark most region of a spike if the background is too noisy. Thereafter, binary masks of spike profiles are seldom used in reality.

For reasons above, spike head voxels are more often marked by experts as ground truth to localize the spike, followed by alignment and average to generate 3D models for spike. Respectively, our segmentation algorithm also predicts the spike heads, even though it takes similar time cost in producing the other three representations. In detail, we assign the value calculated by Eq. 5.10 to each voxel, predicting its possibility of being a spike head.

**Evaluation Methodology.** Finally, our segmentation algorithm produce a spike likelihood $\Pr(o'_i = 1|f_i^{A'}, f_i^{C_{se}}, f_i^{C_{sc}}, f_i^{C_{sp}})$ that indicates the posterior probability of a spike head at each voxel. It is crucial to have a proper voxel-wise evaluation for judging the qualities of different segmentation methods. As the spike segmentation is usually followed by aligning and averaging sub-tomograms (3D bounding box) centered at each spike for 3D reconstruction, we formulate spike segmentation as a problem of detecting the spike heads. In general, we need to design a methodology that evaluates detector performance in an unbiased and informative way.

To compare detectors based on different sets of cues, we plotted the number of missed spikes (false negative) against the number of false positives, namely MFN curve. Each point on the curve is generated independently by thresholding on $\Pr(o'_i = 1|f_i^{A'}, f_i^{C_{se}}, f_i^{C_{sc}}, f_i^{C_{sp}})$ to produce a 3D binary segmentation mask and then matching it with the ground truth. This is preferred to traditional precision-recall curves as it is easier for the biologists to set an upper bound on the

accepted number of false positives independent of the spike density. However, to compute false negative and false positive is not straightforward. One could simply count the overlaps between the 3D binary segmentation mask and the ground truth as true positives, declaring all unmatched voxels either false positives or false negatives. However, this measure is unable to tolerate minor localization errors even though the algorithm may generate useful detection results that may be 1 to 2 voxels away from the ground truth. In fact, due to extremely low SNR, most of the faint targets are not annotated at the exact location in the ground truth by human experts. Thus a measure that tolerates minor localization difference is necessary for evaluating nano-scale semantic segmentation algorithms.

For the above issue, we perform tomogram evaluation using a modified version of the evaluation protocol in the PASCAL visual object detection challenges [39]. A detected voxel and a ground-truth voxel form a potential match if their Euclidean distance is smaller than a threshold $d$, namely *matching distance threshold*. Each detected voxel and ground-truth voxel may be matched at most once. If a ground truth voxel matches several detected voxels, the potential match with the smallest distance is counted as true positive (ties are broken arbitrarily). Finally, unmatched ground truth voxels were counted as false negatives whereas unmatched detected voxels were counted as false positives.

Besides, we use the *average miss rate* (AMR) for each MFN curve to summarize the performance of each detector, approximated by averaging the miss rate at 31 MFN rates evenly spaced in the range 0 to 300 false positives. For curves that stop before reaching a given MFN rate, the smallest miss rate achieved is used. This measure is intuitively similar to the *average precision* in PASCAL challenge because the entire curve was described by a single value. As the number of ground-truth voxels is quite small in contrast to the number of voxels in the entire tomogram, our measure in terms of the average miss gives a more informative and stable illustration of the performance.

### 5.4.2 Detection Accuracy

Given the evaluation methodology above, we need to compare our voxel-wise segmentation algorithm with a baseline algorithm. However, as explained previously, the low SNR and the large size yield difficulty in using most up-to-date segmentation methods. It is hence very challenging to construct a good baseline technique for comparison. To show the contribution of context cues

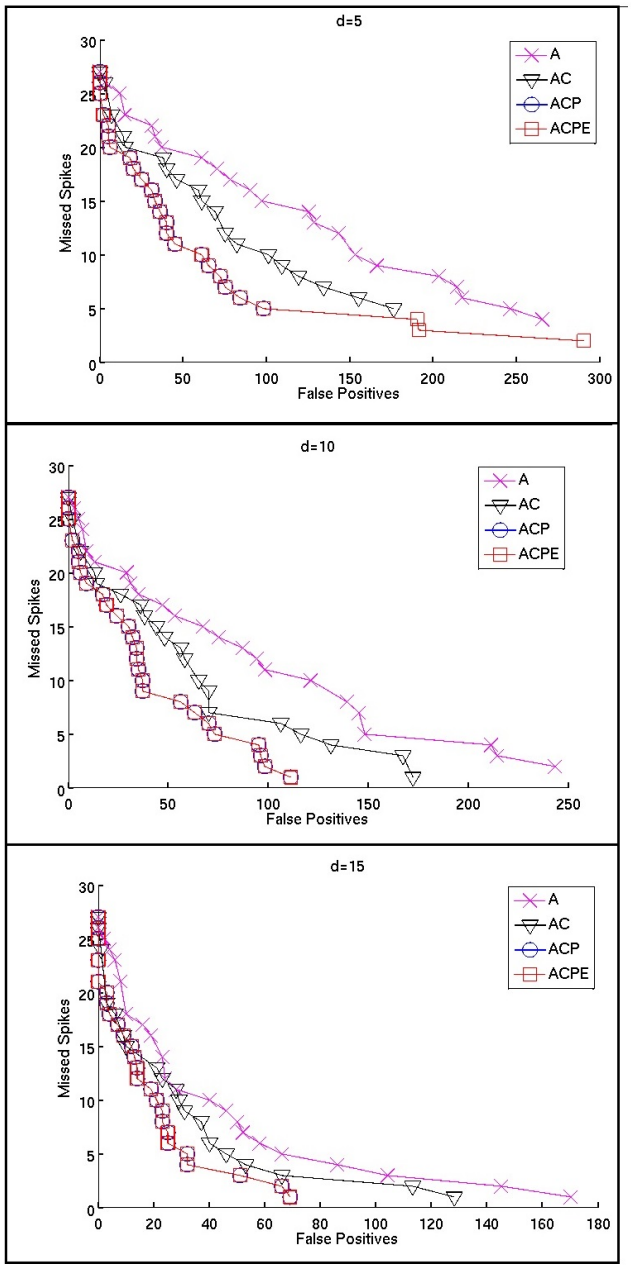



Figure 5.3: Spike head segmentation performance on the microvillus tomogram for different value of matching distance thresholds $d$'s, by thresholding the baseline object-centered model (magenta crosses), the complete context-sensitive model (red squares), and its two ablation models (black triangles and blue circles). Our context-sensitive models yield significantly better performance than the baseline model and our complete model achieves the best performance at all values of $d$. See the text for a description of each model.

Table 5.1: Average miss rate for our model (4), the ablations of our model (2,3) and a baseline appearance-based model (1) with 3 different $d$'s. Our context-sensitive models outperform the baseline model for all values of $d$. See the text for a description of each model.

| Model | $d = 5$ | $d = 10$ | $d = 15$ |
|---|---|---|---|
| (1) A | 0.46 | 0.39 | 0.26 |
| (2) AC | 0.36 | 0.28 | 0.20 |
| (3) ACP | 0.30 | 0.20 | 0.18 |
| (4) ACPE | 0.25 | 0.20 | 0.18 |

in context-sensitive semantic segmentation, we apply thresholding on (5.1) as a baseline algorithm that is purely based on the appearance feature.

Table 5.1 shows the performance for each model to demonstrate the improvement due to different types of context cues. Model names are shown as follows: (1) is our baseline object-centered segmentation, (4) is our complete model, and (2) and (3) are incomplete models using subsets of the context cues in (4). The model name implies what features are used: 'A' for appearance feature in Section 5.2, 'C' for scale context feature in Section 5.3.1, 'P' for spatial context feature in Section 5.3.2, and 'E' for semantic context feature in Section 5.3.3 (here $\lambda = 0.8$). We also present Fig. 5.3, which plots the number of missed spikes against the number of false positives at different matching distance thresholds. In Fig. 5.4 and Fig. 5.5, we visualize two exemplar slice cuts of several sample output spikes of several models, along with the ground truth annotation in the cropped original tomogram. In Table 5.3, we present the performance for each model on the HIV data set.

As shown in our results, the helpfulness of different types of context cues is quite clear. The baseline algorithm based on appearance feature does a worse job in contrast to algorithms with additional context cue(s). In addition, our complete model outperforms each algorithm that uses a subset of all the three context cues. Thus all types of context cues contribute helpful and complimentary information for context-sensitive semantic segmentation. By thresholding our complete context-sensitive model, we have the spike heads, based on which we can generate the respective spike roots with the method mentioned in the spatial context. As the general shape of small spikes is close to a cylinder, it is easy to generate the ridge of each spike given its head and root, or even

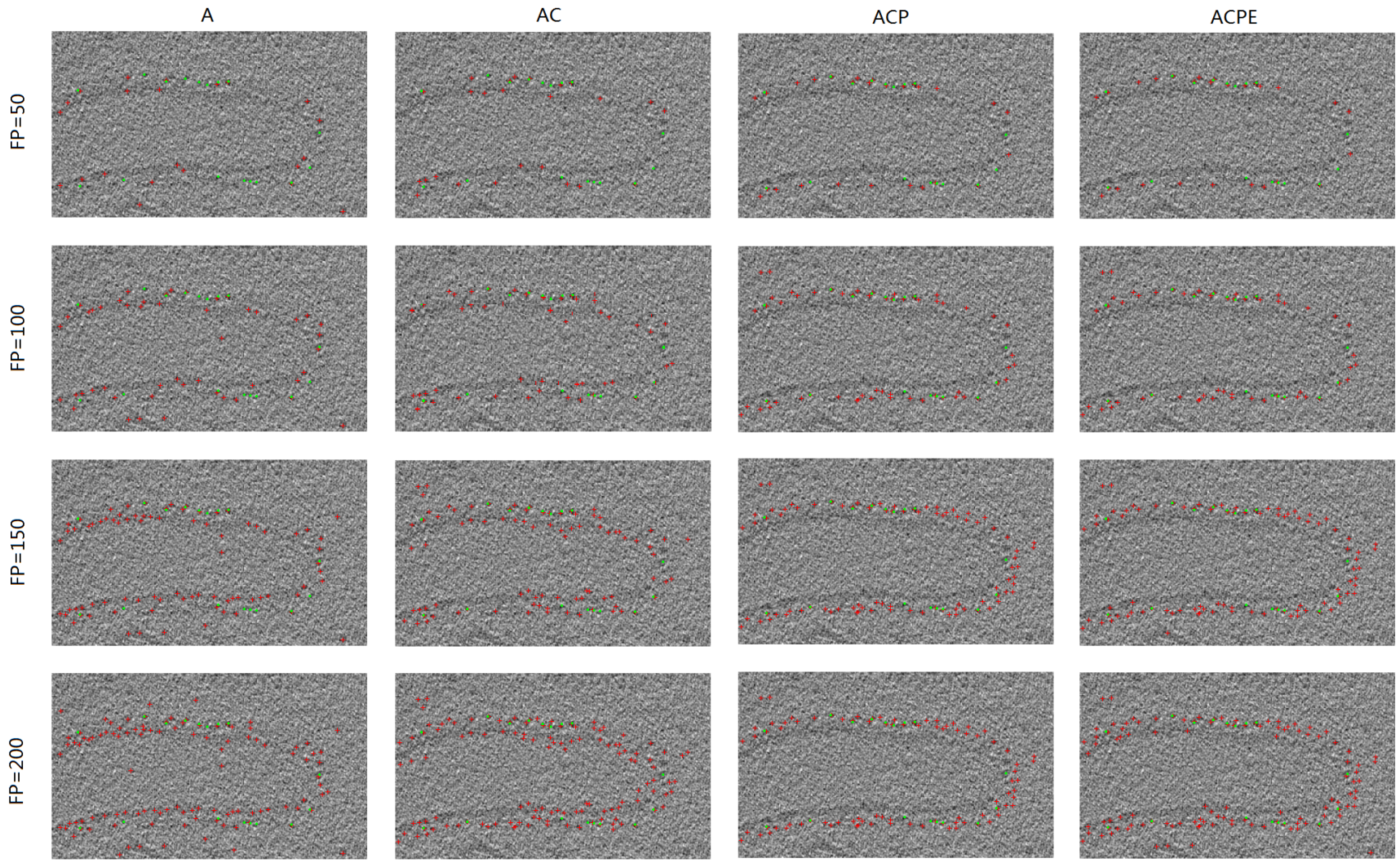



Figure 5.4: Visualization of spike head segmentation on one exemplar slice of the microvillus tomogram. The green dots are the ground truth. The red crosses are the spike heads detected by the respective model. See the text for a description of each model.

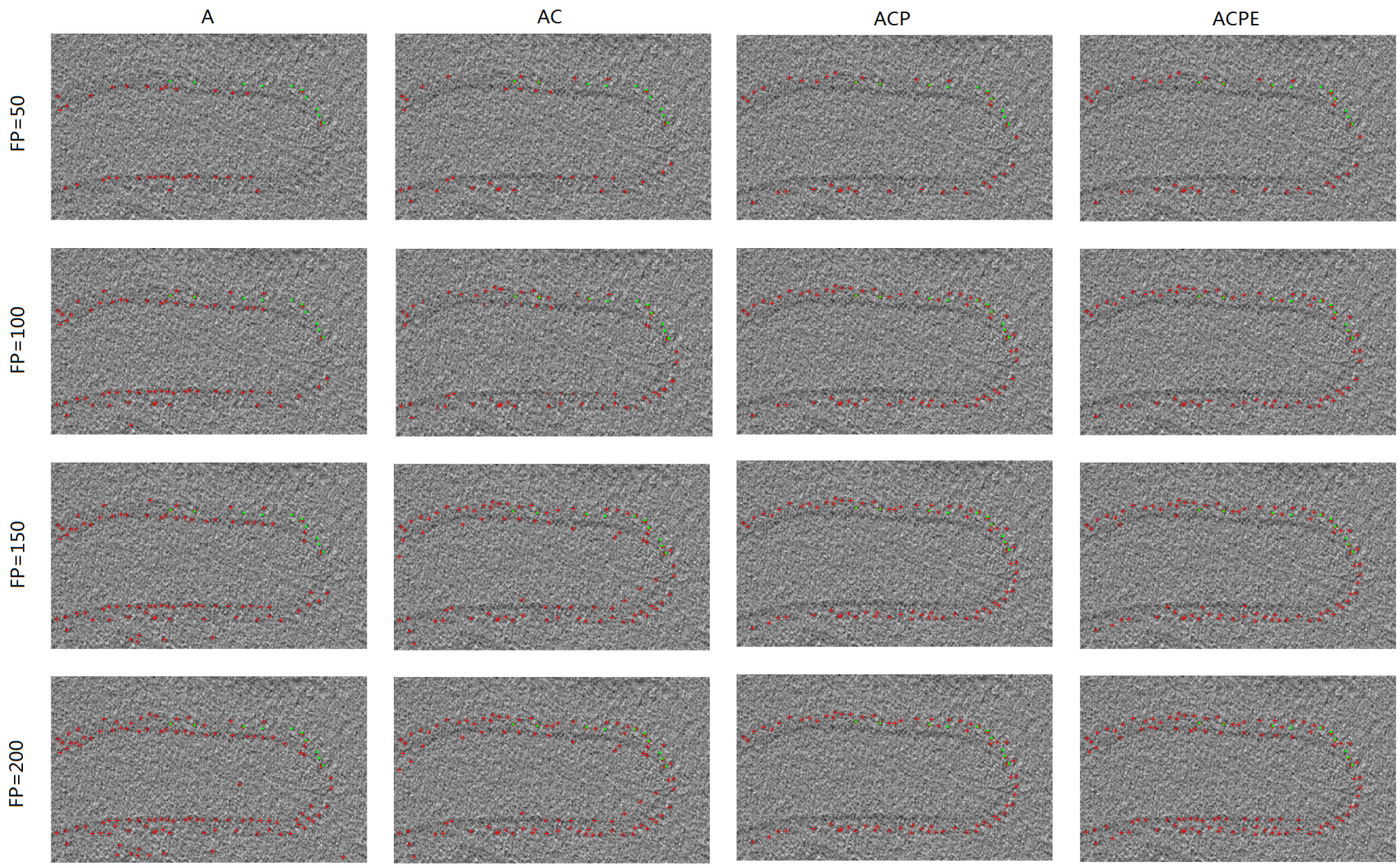



Figure 5.5: Visualization of spike head segmentation on another exemplar slice of the microvillus tomogram. The green dots are the ground truth. The red crosses are the spike heads detected by the respective model. See the text for a description of each model.

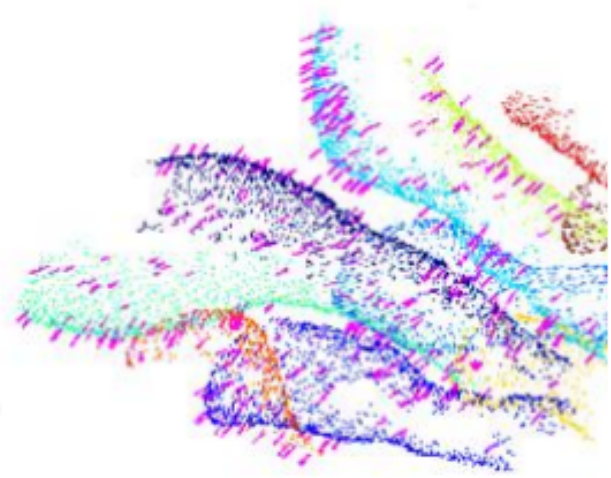


Figure 5.6: Illustration of a sample spike segmentation visualized in 3D, where each magenta segment represents the ridge of a spike.

the sub-tomogram centered at the center of each spike given the general width of a spike. Figure 5.6 illustrates a sample 3D view of our segmentation in which the ridge of each spike is represented as a magenta segment. Note that the result is in 3D and they can thus be visualized from different view angles. By tuning the threshold based on the observation of the segmentation result and other prior knowledge, researchers are able to control the number of potential spikes in an interactive manner, which is convenient for further processing such as averaging. Thus our method provides an efficient tool of nano-scale small object segmentation with great flexibility for researchers.

### 5.4.3 Choice of Context Sensitivity Coefficient

As the proposed complete model involves coefficient $\lambda$ describing the sensitivity of the semantic segmentation to context, we also designed an experiment to show the influence of $\lambda$ on the segmentation performance (the average missing rate). In this experiment, $\lambda$ was sampled by dividing the interval [0, 1] into 51 equal parts. Fig. 5.7 shows the average miss rate of our complete model (d=5) as a function of $\lambda$. We have two observations: 1) there is an optimal solution between object-centered segmentation and context-integration model, indicating a trade-off between the false negatives caused by appearance and context cues; 2) such optimized performance nearly

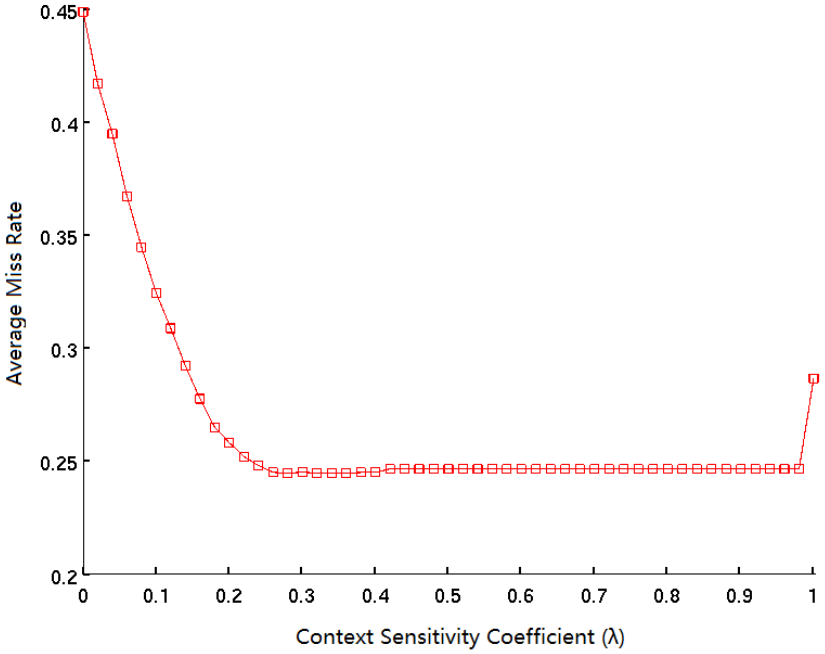



Figure 5.7: The segmentation performance (average miss rate ) of our complete model for different values of the context sensitivity coefficient ($\lambda$). The horizontal axis represents $\lambda$ in our complete model (5.10) and the vertical axis shows the segmentation performance. Our model achieves the best performance (0.2447) at $\lambda = 0.28$.

holds for a wide range of $\lambda$, the lower bound of which naturally and explicitly describes the context sensitivity of semantic segmentation in terms of specific problems.

### 5.4.4 Computational Complexity

In terms of computational cost, our method is much faster than annotating by an expert, spending around 1˜2 hours and marking only 89 spikes for the given tomogram. Table 5.2 summarizes the time cost for each step of our model and the total timecost, which is 3 to 6 times faster than manual annotation. Note that, in the case of annotating thousands of spikes in this tomogram, the time required for the human annotation would be an order of magnitude higher than our method.

## 5.5 Experiment on HIV Tomogram

In this experiment, we evaluated our semantic segmentation method on a 3D tomogram of HIV. The performance of our method was assessed on the task of HIV spike head detection.

Table 5.2: Timecost for each step of our method, using 8 threads on the same 64-bit GNU/Linux, and the timecost of annotation by experts.

| Step | | Time (min.) | Multi-threads |
|---|---|---|---|
| Stage 1 | Scale space | 1.40 | No |
| | Local detector | 10.98 | No |
| | Global analysis | 1.17 | No |
| Stage 2 | A | 3.13 | No |
| | C | 0.76 | No |
| | P | 1.58 | Yes |
| | E | 0.62 | No |
| Total time by our method | | 19.64 | |
| Total time by the expert | | 60˜120 | |

Table 5.3: Average miss rate for our model (4), the models using subsets of our context features (2,3) and a baseline appearance-based model (1). Our context-sensitive methods also outperform the baseline model. See the text for a description of each model.

| Model | (1) A | (2) AC | (3) ACP | (4) ACPE |
|---|---|---|---|---|
| AMR | 0.79 | 0.64 | 0.30 | 0.30 |

### 5.5.1 Dataset

The dataset was acquired from the HIV. We have a volume with a size of $864 \times 686 \times 174$, or more than 103 million voxels. Example slice cuts are shown in Fig. 4.1. Similar to the microvilli case, we use 56 spike heads in the tomogram annotated by an expert and the same evaluation methodology to compare detectors based on different set of cues.

### 5.5.2 Detection Accuracy

To show the contribution of context cues in context-sensitive semantic segmentation, we apply thresholding on (5.1) as a baseline algorithm that is purely based on the appearance feature.

Table 5.3 shows the performance for each model to demonstrate the improvement due to different types of context cues. Model names are shown as follows: (1) is our baseline object-centered segmentation, (4) is our complete model, and (2) and (3) are models using subsets of the context cues in (4). The model name implies what features are used: 'A' for appearance feature in Section 5.2,

'C' for scale context feature in Section 5.3.1, 'P' for spatial context feature in Section 5.3.2, and 'E' for semantic context feature in Section 5.3.3 (here $\lambda = 0.8$). We also present Figure 5.8, which plots the number of missed spikes against the number of false positives when matching distance threshold $d = 15$.

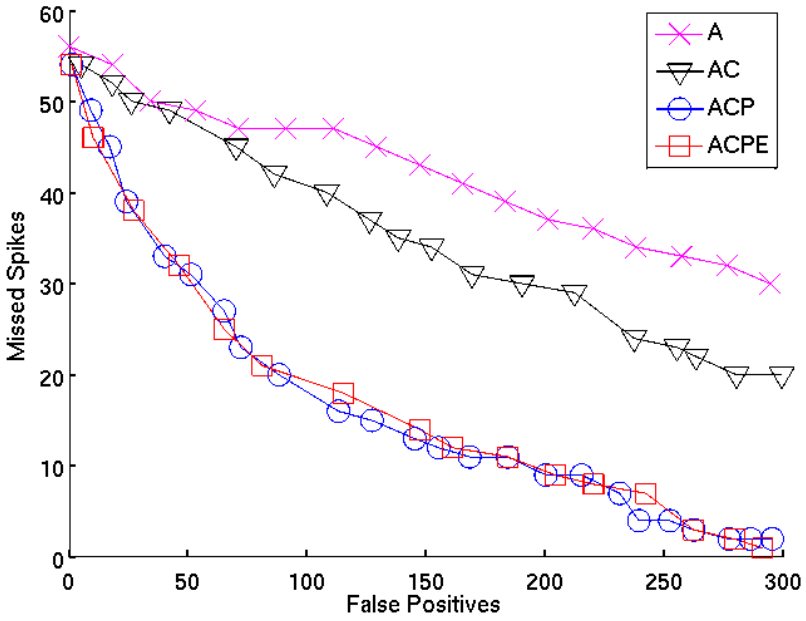



Figure 5.8: Spike head segmentation performance on the HIV tomogram for $d = 15$, by thresholding the baseline object-centered model based on appearance cue (magenta crosses), the complete context-sensitive model (red squares), and two incomplete models (black triangles and blue circles).

As shown in our results, the helpfulness of context cues is also quite clear. The baseline algorithm based on appearance feature does a worse job in contrast to the algorithms with additional context cue(s). In addition, our complete model outperforms each algorithm that uses a subset of all the three context cues. Thus all types of context cues contribute helpful and complimentary information for context-sensitive semantic segmentation.

### 5.5.3 Computational Complexity

In terms of computational cost, our method on HIV tomogram also takes less than half an hour to finish all the steps. In contrast the microvillus tomogram, the number of spikes in the HIV

tomogram is much fewer and the quality of the HIV tomogram is generally much better. Thus the corresponding annotation timecost for an expert is close to the timecost of our method.

## 5.6 Summary

So far, we have presented the use of our novel framework for the problem of nano-scale semantic segmentation, demonstrated on 3D cryo-electron tomogram of spikes on microvilli and HIV. The low SNR and large size of our data make most up-to-date segmentation methods intractable. In contrast, our method has achieved efficient voxel-wise segmentation through context features that do not only tolerate the extremely noisy background, but also reduce the searching space dramatically. We have presented three types of context cues appropriate for nano-scale faint target segmentation – a scale context feature that describes the sizes in which targets are usually found in a scene, a semantic context feature which encodes the co-occurrence of other objects in the scene, and a spatial context feature that offers a specific configuration in which targets and other objects are usually found. Our method models the true posterior probability of a spike head at every voxel, which is significant for further processing such as alignment and averaging. We have also defined a methodology for benchmarking nano-scale segmentation algorithms. Based on a quantitative evaluation on a $600 \times 1400 \times 432$ tomogram with 27 annotated microvilli spikes and a $864 \times 686 \times 174$ tomogram with 56 annotated HIV spikes, we have shown that our complete context model outperforms models using subsets of the context features. Meanwhile, all context-based models are superior to purely appearance-based method concerning the segmentation performance. Therefore, our results indicate that an appropriate treatment of context cues is essential for segmenting objects in nano-scale.

Besides advancing image processing, our work has the benefit of handling a critical and unsolved problem in spike research that of semi-automatically localizing spikes. By producing nano-scale semantic segmentation in an efficient and accurate manner, our method allows spike researchers to focus on the data analysis rather than data processing and thus enables breakthrough biological research at a large scale.

Due to the limited imaging conditions of biological samples, the image quality in nano scale is usually very poor and thus context information often plays a critical role in object identification. To demonstrate our proposed framework is useful for not only nano-scale images but also more

natural images, we will present the application of our framework on the task of tattoo classification based on tattoo image segmentation in the next chapter.

# CHAPTER 6

# CONTEXT-SENSITIVE TATTOO SEGMENTATION AND TATTOO CLASSIFICATION

## 6.1 Introduction

In the previous chapters, we have seen the efficiency and effectiveness of our framework on segmentation of a challenging data type – cryo-electron tomograms. In this chapter, we will show that our framework can also be applied to natural images to improve semantic segmentation. To achieve this, in this chapter, we will apply our two-stage framework on the task of tattoo segmentation, which benefits tattoo classification applications, such as gang identification and tattoo artist identification.

Scars, Marks and Tattoos (SMT) are useful clues for criminal identification and personal identification in criminal conviction and medical forensics respectively. Besides the canonical biometric identifiers such as fingerprint, DNA and iris, a large amount of tattoo images have also been taken from victims, suspects and incarcerated personnel for identification in law enforcement [57]. These biometrics are collected, maintained and analyzed by national security systems like the Integrated Automated Fingerprint Identification System (IAFIS) for retrieval purposes [103]. Manual tattoo searches over a large dataset are very time-consuming and inefficient. Several Content-Based Image Retrieval (CBIR) systems have been proposed for tattoo matching and retrieval [57, 58, 1, 71, 52]. The performances of these systems are sensitive to tattoo segmentation, which is a pre-processing step to remove varied background. Our goal in this chapter is to demonstrate the effectiveness of our context-sensitive framework on natural images by designing a tattoo segmentation system to automatically mark tattoo regions.

The objective of tattoo segmentation systems is to extract regions solely containing component(s) of tattoos in an image. Tattoo segmentation requires that each extracted region has a semantic component of tattoo. This is more than traditional bottom-up image segmentation [30, 130, 22, 121], which only requires that each segmented region be homogeneous, known as over-segmentation. As bottom-up segmentation is sensitive to intra-object variance, various forms of

top-down cues are usually combined with bottom-up cues for the purpose of obtaining semantic meaningful results. For example, Schitman et al. [118] found that a group of patch sets (one for each class and labels are known) can help label the homogeneous regions obtained from bottom-up over-segmentation. Each set contains patches sampled from one class in the labeled training image and the cost of the assignment to each class is computed for each over-segmented region. A graph-cut optimization based on these costs is used to find a globally optimal segmentation. Carreira et al. [61] applied multi-scale binary segmentation on an image using the parametric min-cuts technique. Then a feature-based regressor is trained to rank the pool of segmentation results to predict the likelihood of each segment being an object. Such regressor is learned from the statistical distribution of a large number of features (related to graph, region and Gestalt properties) among a set of annotated images. To avoid manually labeling the training set, a large number of images containing the same object were simultaneously segmented in [82], assuming that the common parts of an object will appear frequently while the effect from varied background will diminish. In such an approach, superpixels and interest points are re-organized as mid-level over-segmentation results and visual words representations from bottom-up and top-down priors of a hybrid graph model respectively.

Since bottom-up segmentation is well studied [61, 82, 143, 118], the key of tattoo segmentation is how to incorporate top-down priors. Unfortunately, obtaining top-down priors for tattoo is very challenging due to large variances in tattoo appearance, shapes and spatial connectedness. Each gang has different tattoo patterns with its own symbol system. It could be a particular number, an animal or a combination of several meaningful components. Despite the fact that the number of gangs is limited, the appearance of the tattoo patterns (such as letters, numbers and animals) in a gang still varies from person to person in general. Take the tattoos of a gang named four corner hustlers for instance [1], the tattoo pattern with the semantic meaning of four is shared among the tattoos of this gang. Even though the number four is often observed in these gang tattoos, it is obvious that the appearances (such as texture, color and writing style) of the number fours are significantly different. Sometimes the number one and number four in the tattoo pattern are inked in different fonts. Moreover, the tattoo pattern four may also appear as either the Roman numeral four (IV) or even a diamond with four corners. Such varied appearances induce

[1] Tattoo images are available at `http://gangink.com/index.php?pr=FOUR_CORNER_HUSTLERS/` [48].

large intra-class variance of the tattoo pattern of the gang four corner hustlers. On the other hand, some number fours are even surrounded by characters like C and H indicating corner and hustlers in the gang's name. Such disturbances in the tattoo may also increase the variance of the tattoo appearances of this gang. There also exists a considerable spatial variance of the tattoo patterns. A tattoo may not be a collection of spatially related components, such as rabbit head and letters. Therefore, the neighborhood of components is often difficult to be involved in segmentation model as spatial constraints for region aggregation, which attempts to obtain the object as a whole. Suffering from the shortcomings mentioned above, it is difficult to segment tattoos from the background directly through any prior on location or shape of tattoo as a whole. Thus this problem has been rarely discussed [57, 58, 1]. Jain et al. [58] proposed a tattoo image retrieval system via image matching based on SIFT features. However, in contrast to the performance using their segmentation algorithm, better performance was reported when tattoo images were manually cropped to extract the foreground as well as suppressing the background, which is really time-consuming. Acton and Rossi [1] proposed a segmentation approach based on active contour and vector field convolution. Nonetheless, the contour initialization is difficult to be given when the structure of tattoo is complicated.

Here we propose a tattoo segmentation system combining both bottom-up and top-down priors. We make the assumption that each component (e.g., a letter or a number) in a tattoo is arbitrarily located in spatial space of a tattoo image. Moreover, we consider skin and tattoo as a whole at first and deal with a figure-ground segmentation for both skin and tattoo as the foreground. Figure-ground segmentation is a recognition process that needs to figure out the aimed object(s). Thus top-down priors should be involved for this step. After obtaining regions with skin and tattoo, another figure-ground segmentation distinguishes the tattoo from skin. Similar to the segmentation in the previous step, top-down priors are learned from the image. This two-stage process agrees with the hierarchy and adaptivity of the human visual system for visual scene understanding [120].

The rest of this chapter is organized as follows: Section 6.2 presents the proposed system for tattoo segmentation in details, followed by experimental results. In section 6.3, a novel gang identification system is proposed based on tattoo segmentation in section 6.2, along with a comparison between the tattoo recognition with and without our tattoo segmentation results. A discussion on some further improvements and a conclusion are given in Section 6.5.

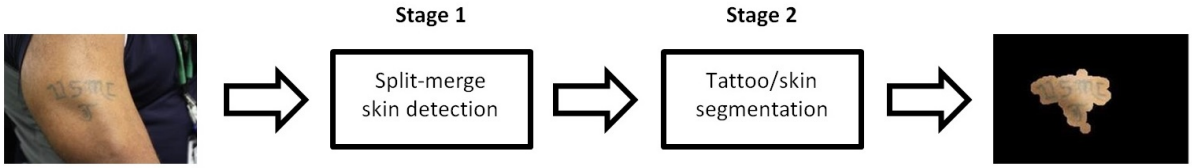



Figure 6.1: The outline of context-sensitive tattoo segmentation.

## 6.2 Context-Sensitive Tattoo Segmentation

Regarding the complexity and intra-class variance of tattoo, our main idea is to transfer the tattoo segmentation into skin detection followed by a figure-ground segmentation. The outline of our algorithm is depicted in Fig. 6.1. First, a clustering technique is used on the range domain (color space) to separate the tattoo image into over-segmented regions in a bottom-up manner. This is a very important step in that regions containing both skin and tattoo are much more non-homogeneous than these over-segmented regions. Then, based on top-down cues learned from the image itself, a region merging step is introduced to group skin regions together. Through this split-merge process, skin and tattoo are distinguished from the background. Finally, K-means algorithm is applied for figure-ground segmentation, where now the tattoo is the foreground and the skin is the background.

### 6.2.1 The First Stage: Split-Merge Skin Detection

**Preprocessing.** In skin detection, an image is usually regarded as a group of feature vectors. Each pixel corresponds to a feature vector $f_i^A$ in a multi-dimensional feature space. The statistical properties of the histograms or the distributions built on these dimensions are widely discussed [62, 129]. To simplify our system, we begin by a preprocessing step that represents an image as a set of homogeneous regions in terms of certain properties, namely superpixels, $I = \cup S_i$. Here $i = 1, ..., N$ and $N$ is the number of superpixels. Specifically, an initial clustering process is carried out on the gray-scale distribution $h(I)$ of the image. Let $\{f_i^{A_g}\}$ be the set of the intensity values from the gray-scale image of $I$, while the function $h(x)$ evaluates the density estimate covering the range of $x$. Since pixels from the same cluster are more likely to have a similar intensity, we segment the tattoo images according to the local minima of the gray-scale distribution. In that sense, the

pixels with intensity values between either two closest local minima are labeled as a cluster. To this end, other clustering algorithms like watershed and mean-shift could be used as alternatives. After applying the histogram-based clustering, $d1$ local minima in the gray-scale distribution produce $d$ clusters in the gray-scale space. Obviously, the resulting initial segmentation (superpixels) may suffer from under-segmentation due to background with similar intensity and over-segmentation due to illuminant variations on the skin. However, the following steps can help alleviate such problems:

**Initialization.** Since only weak prior information in detecting tattoos can be obtained in a given image, the segmentation system needs to be initialized, where an initial model of skin (with tattoos) needs to be estimated. Based on the observation that the center region in almost all the tattoo images in the database contains skin, we use a center in focus initialization, in which the statistical properties of the central region are considered as prior knowledge for skin detection. Clearly, this initialization step is not universal and could be replaced if some prior information about the skin color is available.

**Connected Structure.** Although skin with tattoo may be split into several regions (clusters) due to shading and intra-class variance of tattoo components, these regions are more likely to be connected to each other in spatial space. If the main pattern of tattoo is not in the center of an image, it may still be merged since its background in the cluster (the skin) may be adjacent to the pure-skin region in the center. Therefore, it is reasonable to use the neighborhood of clusters in spatial domain as a top-down cue for merging potential skin regions.

Following these two points above, an $m \times n$ patch $p_0$ in the center of $I$ is sampled from $I$ for obtaining prior knowledge (empirically, $m$ and $n$ is half of the height and width of I correspondingly). How to obtain the seeds for learning the top-down prior of the objects is widely discussed in figure-ground segmentation [61, 82]. Joao et al. [61] randomly pick up foreground seeds from a group of pixels uniformly distributed over the spatial space for several times. The background seeds are those on the boundary of an image. Liu et al. [82] use interest points obtained from a large amount of images containing the same object as the prior for such an object. Regarding our initialization, clusters covering the major area of $p_0$ are most likely to be skin and tattoo. Thus in our work, a region sampled from the image is regarded as the seeds containing top-down priors. In detail, the number of overlapped pixels between $p_0$ and each cluster $c_i$ are sorted as $\{(n_j, i)|n_j > n_{j+1} \text{ and } i, j = 1...d\}$. The first $k$ clusters with the largest number of overlapping

pixels are labeled as potential foreground (skin and tattoo) under the constraint that their overlap ratio $\frac{\sum_{l=1,\ldots,k} n_l}{\sum_{l=1,\ldots,d} n_l}$ exceeds a threshold t (typically 75%). In most cases, such potential foreground may contain arbitrary number of regions merged by clusters. However, only the region with overlap in the sample patch is labeled as foreground due to the connected structure of skin. Therefore, only one region with sample patch inside is segmented as skin. If either skin or tattoo dominates the sample path, regions belonging to the other may be excluded from the segmented region. Thus an operator filling the holes inside such regions should be applied as a post-process of skin detection.

### 6.2.2 The Second Stage: Figure-Ground Tattoo Segmentation

To this point, we transferred a problem of tattoo segmentation with unknown number of clusters to a skin-tattoo binary segmentation, where skin is the context of tattoo. This is based on the spatial context that tattoos almost always appear on skins. In this section, skin pixels should be distinguished from pixels belonging to tattoos. In that sense, a skin pixel in this section indicates merely the skin pixel that is not covered by tattoos. Since we already know the number of potential clusters now, a k-means algorithm ($k = 2$) can be applied on the RGB color space of the foreground (skin). Now the issue is which cluster should be tattoo. If distinction between tattoo and skin is needed, the pixels on the contour of the skin region are more likely to be skin pixels rather than pixels in tattoo. Because, otherwise, skin is fully covered by tattoo and distinguishing tattoo from skin is thus unnecessary. Following this rule, the cluster with more pixels on the contour of the foreground is labeled as the skin and the other the tattoo. If the structure of tattoo is preferred rather than the whole region containing tattoos in applications, an alternative way of marking the tattoo is to apply a ridge or edge detector [81, 25] on the skin region. This is reasonable since tattoo is a kind of man-made painting with clear boundaries while skin regions are textureless in contrast.

### 6.2.3 Experiments

**Experiment on a single tattoo from different views.** First, we have tested our algorithm on a single tattoo taken from different views. As shown in the first column of Fig. 6.2, the images of a military tattoo were taken from different views. Our context-sensitive segmentation captures the military tattoos accurately regardless of the views.

Figure 6.2: Our segmentation results of a tattoo from different views. Each row is one view. The first column shows the original images, whereas the second shows our segmentation results.

Table 6.1: More details of the accuracy and the $F$ measure of proposed algorithm.

| | Min | Max | Mean | Variance |
|---|---|---|---|---|
| Accuracy | 0.7989 | 0.9614 | 0.8983 | 0.0014 |
| F measure | 0.4204 | 0.7854 | 0.5866 | 0.0061 |

**An Experiment on a Tattoo Database.** We have also tested our proposed algorithm over a collection of 256 tattoo images [2]. Each tattoo is unique in the sense that no two images were taken from the same tattoo under different views or illuminant conditions.

Figure 6.3 shows the accuracy distribution of the proposed algorithm. Here the segmentation accuracy is the most widely used evaluation metric defined as follows:

$$Accuracy = \frac{|S_f| + |S_b|}{|S|}, \tag{6.1}$$

where $S_f$ and $S_b$ are correctly assigned foreground and background pixels correspondingly and S is the image. $|X|$ means the number of data (pixels) in the set $X$. Since bad segmentation may receive a good accuracy if tattoo is small, as suggested by Liu et al. [82], our algorithm was also evaluated under a popular measure for information retrieval, called F-measure:

$$F_\alpha = \frac{(1+\alpha) \cdot R_r \cdot R_p}{\alpha \cdot R_p + R_r}, \tag{6.2}$$

where $\alpha$ is a balance parameter for precision

$$R_p = \frac{|A \cap B|}{|A|} \tag{6.3}$$

and recall

$$R_r = \frac{|A \cap B|}{|B|}, \tag{6.4}$$

where $A$ indicates the man-made ground truth segmentation and $B$ the result of proposed segmentation algorithm. $\alpha$ is usually set as 2. Its distribution is shown in Fig. 6.4. More details of the accuracy and the $F$ measure of proposed algorithm are shown in Table 6.1.

[2]Images are available at `http://ganginк.com/` [48].

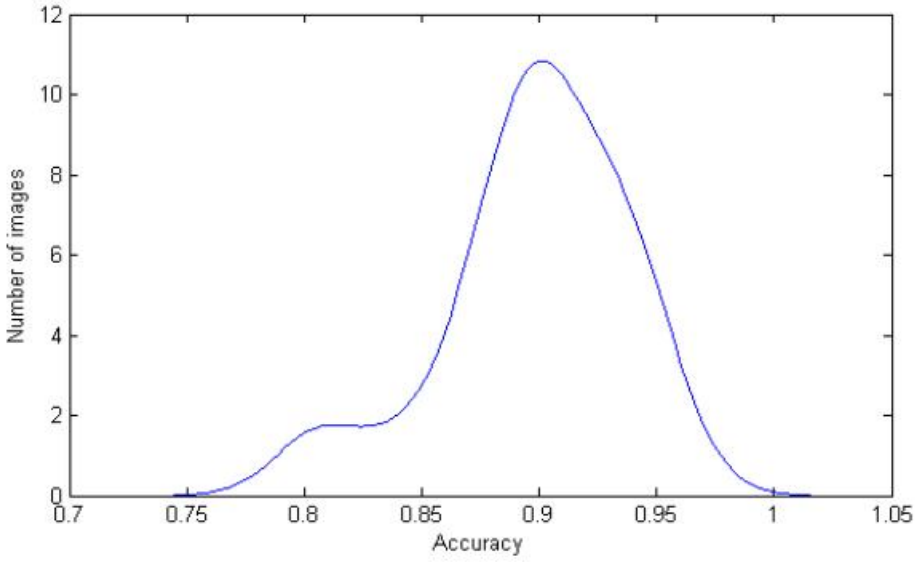



Figure 6.3: The accuracy of our algorithm. The $x$ axis is the accuracy and the $y$ axis is the number of images involved in each accuracy.

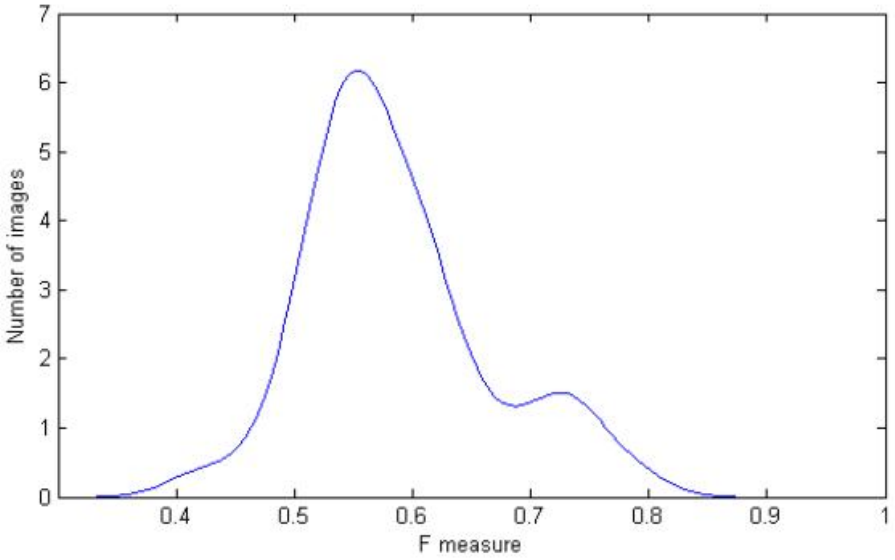



Figure 6.4: The $F$ measure of our algorithm ($\alpha = 2$). The $x$ axis is the $F$ measure and the $y$ axis is the number of images involved in each $F$ measure.

Our experiment shows that our algorithm can separate the tattoo regions from most of the backgrounds although in some examples certain small tattoo regions are further eliminated. This is not a big problem for tattoo classification since these regions are small in contrast to the whole tattoo and the main patterns of the tattoo are still well reserved.

## 6.3 Tattoo Classification

### 6.3.1 Motivation

Human adoption of SMT was typically observed in two groups: military and gangs. Some gangs are closely affiliated to terrorist groups, such as MS-13 (US) & MS-18 (San Salvador) in Mara Salvatrucha and PEN1 in Public Enemy Number 1. Prison gangs also cause trouble such as strong-arm extortion and conflicts between gangs in prisons. Thus the identity encoded in gang tattoos can be a matter of life or death. On the other hand, in the past ten years, we have also seen a large increase in the adoption of tattoos by mainstream media (i.e., movies *Blade* and *Pop-Culture*) and wider population. As reported by Laumann et al. [69] in 2006, 24% of people aged 18 to 50 in the USA have at least one tattoo, and this number is still increasing. Thus there is a growing demand in building a gang identification system based on tattoo classification to increase the possibility of preventing potential violence and crime.

There are only a few researches in the computer vision community that are close to our task. Jain et al. [57] proposed a tattoo image matching system, called Tattoo-ID, to find the most similar tattoo images in the database. This system extracts interest points via scale invariant feature transform (SIFT) and then measure the distance between the query tattoo image and every image in the tattoo image database by an unsupervised ranking algorithm based on the points. The performance of this work was improved in [71] by developing more robust matching metric and using a forensic oriented image database with metadata. In a similar work, Acton et al. [1] proposed another tattoo image matching system by active contour and global-local image features (i.e., color and shape). Han et al. [52] extended the tattoo image retrieval from the image-to-image matching to the sketch-to-image matching. However, all the works above suffer from the fact that the input images must be manually cropped to remove background noise beforehand. In addition, given some distance metric that measures the similarity between images, all these works are aimed at finding the most similar tattoo images in the tattoo image database. Thus their application on

gang identification is limited by their assumption that the query image is close to a duplication of some images in the database. To the best knowledge of ours, no work has been reported for tattoo classification based on automatic tattoo segmentation.

Even though a tattoo usually consists of arbitrary number of connected components, the typical distinctive components between gang tattoos are often one or two particular patterns embedded in the tattoos. They usually appear as semantic level patterns, such as number four and rabbit head. We call them *tattoo patterns*. In other words, tattoo patterns are the connected components in a tattoo with special structure contributing to gang identification. Figure 6.5 shows the connected components for some tattoos based on the ridge detection of our segmented tattoo regions. The tattoo patterns like rabbit head, numbers and human head pop up in terms of connected component with different colors. Despite the difficulty of representing the entire tattoo in an image as a meaningful structure, tattoo classification based on shape analysis is reasonable when taking tattoo patterns into account. Therefore, in contrast to find a "near duplicate" image from the database, it is more reasonable for gang identification to consider a matching between the tattoo pattern sets from the query image and the images of known gangs. Following our context-sensitive tattoo segmentation system, it is straightforward to further represent tattoos in the query and the database as their patterns, in the form of connected components. In our work, we rely on advanced data structures to support efficient large scale search. It allows the identification of the gang to which the tattoo at hand belongs based on matching features with all known and new gang tattoo patterns. In what follows, I will describe the details of our gang identification system based on tattoo classification.

### 6.3.2 Tattoo-Based Gang Identification

**Tattoo Dictionary.** For identification purposes, tattoo patterns are normally distinguishable among the connected components due to their better connectedness. Therefore, it is reasonable to sort the connected components regarding their sizes and label the largest $k$ components as potential tattoo patterns for further analysis. Given the segmentation of a set of tattoo images from a target group (i.e.: tattoos for a specific gang or a specific artist), we apply morphological filling on each connected component and then generate a collection of the potential tattoo patterns, namely the tattoo dictionary. Meanwhile, we will also generate the potential tattoo patterns from the query tattoo image.

Figure 6.5: Illustration of connected components in tattoos. The first row shows the tattoo segmentation results from the ridge-based descriptor and the second row shows different connected components associated with different colors.

**Feature Extraction.** To include global information of tattoo patterns for classification, we have developed algorithms for producing both appearance-based descriptors (scale invariant feature transform, or SIFT) [85] and shape-based descriptors (Shape-DNA) [112]. The former is a vector showing the statistics of a semi-global region centered around a point which is given by a detector. The point is usually given by some widely used interest point detectors, such as Harris interest point detector [53] and SIFT detector [85]. Harris interest point detector is affine-invariant and marks a set of points in an image with local maxima of the cornerness. Here corners are defined as locations where the image signal varies significantly in both directions and thus cornerness reflects how much the variation is. However, it is sensitive to change in image scale. As reported by Moreels and Perona [97], the combination of Harris detector and SIFT descriptor is best for lighting changes. On the other hand, SIFT detector marks a set of points in an image with scale-space extreme. Thus it is scale-invariant. In contrast to SIFT, Shape-DNA is not only scale invariant but also reserves unique shape signature. Moreover, it dramatically reduces the discriminative information needed to be stored. It is the eigenvalues (i.e. the spectrum) of a Laplace-Beltrami operator in terms of a given shape:

$$\Delta f = div(\nabla f), \tag{6.5}$$

where $\nabla$ is the gradient and $div(x)$ is the divergence of $x$ on the manifold. Due to the merits mentioned above, our use of Shape-DNA from potential tattoo patterns has great potential in

outperforming the use of SIFT for gang tattoo classification.

### 6.3.3 Experimental Results

**Basic Strategies.** To focus on the performance of different features used, the nearest neighbor (NN) search is used as the classification strategy in this part. For each tattoo image, we set $k = 5$ and thus the largest five potential tattoo patterns are collected for feature extraction. To test the performance of our proposed method, we adopt the commonly used leave-one-out testing procedure. Let $N$ be the number of images with known label (known class) and each of them contains a number of SIFT descriptors. Each time, we leave one image out as the query image and use the remaining $N - 1$ images in the database as the training set. To each SIFT descriptor of the query image, we find in the training set its closest SIFT descriptor, named a match. The test image is classified as the cluster with most matches. The leave-one-out testing procedure continues until all the $N$ images are left out once. The significant difference between the task of tattoo image retrieval and ours is that, rather than image-oriented, our work is SIFT-descriptor-oriented. In contrast to return the most similar training images, our approach seeks for the most similar SIFT descriptor (a match) in the whole training set for each SIFT descriptor from the query image, followed by classifying the query image based on a SIFT-feature (match) voting. The query image is classified as the cluster with most matches.

**Professional and Non-Professional.** Since professional tattoos are usually more complicated than non-professional tattoos and SIFT point detection can thus be time-consuming, the Harris interest point detector and SIFT descriptor are used. We have applied our classification on a set of 40 tattoo images, 20 with professional tattoos and 20 with non-professional tattoos. Our experiment has achieved **85%** accuracy of distinguishing professional tattoos from non-professional tattoos.

**Gang and Non-Gang.** Since gang tattoos contain certain patterns which may appear in different scales, scale-invariance is more important than the previous problem. Hence, to distinguish gang tattoos from non-gang tattoos, here we have used both SIFT detector and SIFT descriptor. We have applied our classification on a set of 40 tattoo image, 20 with gang tattoos and 20 with non-gang tattoos. Our experiment has achieved **77.5%** accuracy of distinguishing gang tattoos from non-gang tattoos.

**Continent United States Gangs (CONUS) and Outside the Continent United States Gangs (OCONUS).** CONUS and OCONUS have similar agendas such as extortion, murder, drug trafficking and terrorist affiliation. Meanwhile, SMT are used heavily by both. However, CONUS have their origins in American Penal Systems, whereas OCONUS are politically tied to an impacted Social Group. Thus it is also necessary to design a classifier for distinguishing tattoos from these two groups. Due to limited number of SIFT key points from tattoos of these two gangs, we have used Harris interest point detector instead, followed by SIFT descriptor. We have applied our classification on a set of 40 tattoo image, 20 from CONUS gangs (such as Crips, Bloods) and another 20 from OCONUS (such as MS-13 and Russian Mafia). Our method has approached the accuracy of **70%**.

**Sets and Sub-Sets.** Sets here are typically sub-sets of a known gang separated by streets, cities, and states. In fact, set on set violence is higher than gang on gang violence. Thus it is also necessary to design a classifier for distinguishing tattoos from these two groups. Similarly, SIFT key points and SIFT descriptors are used. We have tested our method on 15 tattoo images from AB and 15 tattoo images from MS-13&18. Our experiment has achieved **73.33%** accuracy of distinguishing MS-13&18 from AB.

**Signature of Artist.** Even though the SMT may be different, the artist that created them may be the same. Sets typically have an internal artist do their work. Thus the classification of the tattoo based on the intrinsic properties of the artist may also allow traceability and affiliation of members. Similarly, SIFT key points and SIFT descriptors are used. We have applied our classification on a set of 30 tattoo images from three artists (10 images from each). Our experiment has achieved **95%** accuracy of style identification of three artists.

**Gang-tattoo Identification.** Tattoos from each gang, in most cases, have their own tattoo patterns with distinguishable shapes, rather than appearance, which can be identified by a unique shape signature called Shape-DNA. Therefore, our gang-tattoo identification is based on Shape-DNA instead. Our experiments on a database containing tattoos from both 12th Street Players and Familia Stones have shown that the performance of Shape-DNA on gang-tattoo identification is better than that of SIFT, which is **81.25%** versus **62.5%** in accuracy. Figure 6.6 illustrates some examples of the largest potential tattoo patterns for extracting the Shape-DNA. The first two rows

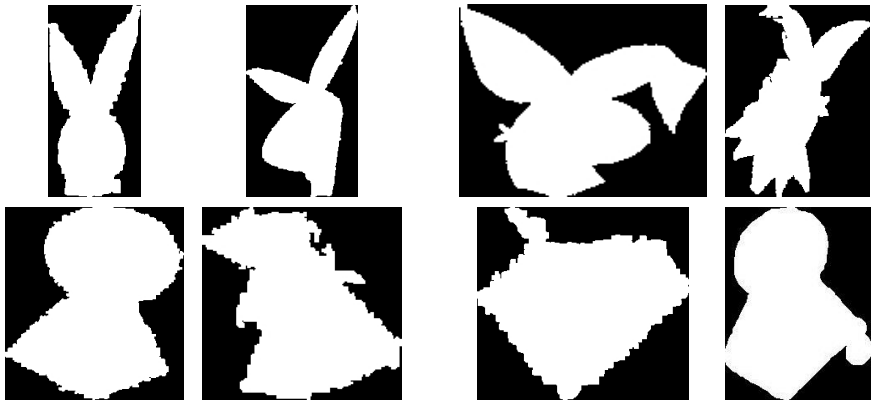


Figure 6.6: Some examples of the largest potential tattoo patterns extracted from the tattoos of 12th Street Players (the first row) and Familia Stones (the second row) for producing the Shape-DNA's.

of Fig. 6.7 are shape-DNAs for the first two patterns in the first row of Fig. 6.6, whereas the last two rows are shape-DNAs for the first two patterns in the second row of Fig. 6.6 respectively.

## 6.4 Discussion

Given the tattoo segmentation of an image, we can also represent the shape of each connected component by its ridges. By computing the eigenvalue of the Hessian matrix [81], ridges appear to be the local extreme pixels along the largest surface curvature. The third row of Fig. 6.5 illustrates some example of the ridge responses deriving from the segmented regions. After applying the ridge detector, a set of images with connected components can be collected. Each connected ridge can be regarded as a potential tattoo pattern. In other words, each tattoo is cropped into finer components with less semantic meaning but clearer shape structure. Based on those unit connected components, the shape-like features and the ink style features can be modeled separately for each part of tattoos. After that, the database is built up by those connected components based features. For ink style features, it will be gradient-like features (measuring the sharpness of the lines) and kernel-density estimation of the color distributions. For shape-like features, there are a number of choices such as spectral histogram features [84], shape context [11] and other features that are more sensitive to shapes. For shape context, we can borrow the method from the previous work in [12]. This kind of shape recognition has been applied well to recognize silhouettes, trademarks,

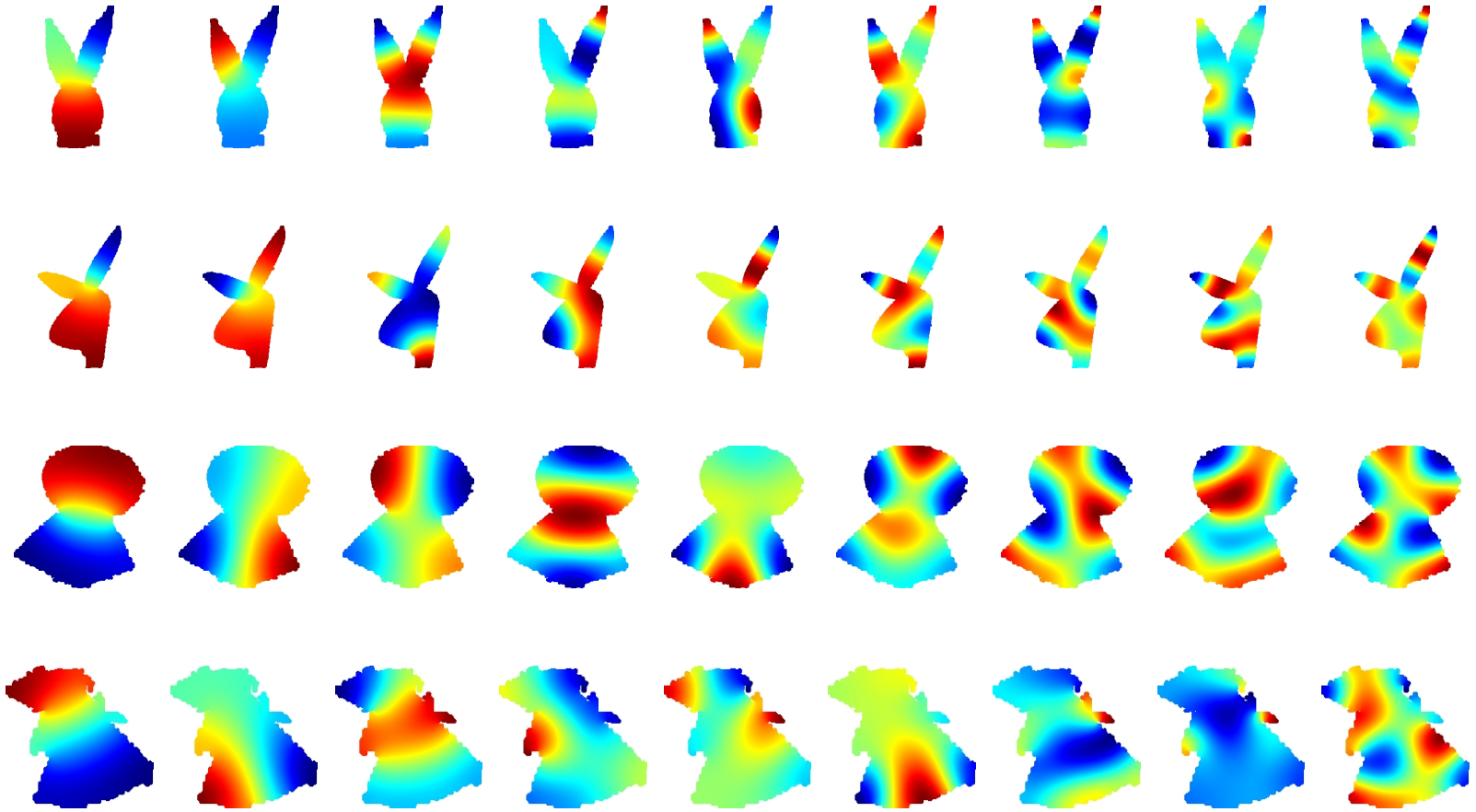


Figure 6.7: The shape-DNAs of potential tattoo patterns. The first two rows are shape-DNAs for the first two patterns in the first row of Fig. 6.6 respectively. The last two rows are shape-DNAs for the first two patterns in the second row of Fig. 6.6 respectively.

and handwritten digits. Since the similar properties between tattoos and those types of data, the shape context features are intuitively a strong potential method for tattoo. Given a new query tattoo image, the steps above can be repeated to generate a set of features based on connected components. Through similarity measurements, we can label the query tattoo by one of the known classes in the database. Despite the simplified design of the algorithm proposed as a prototype system, its process is quite general for tattoo segmentation and the performance of each step can be improved by more effective methods such as those mentioned in each step.

In terms of the classification strategies, there are also alternatives such as shape matching method based on sliding wavelets [106] and approximate nearest neighbor (ANN) searching approaches [10]. The classification performance of the former was reported above 85% on shape and thus has great potential in tattoo-shape classification. For a given expected nearest neighbor matching accuracy of 90%, for a 1M SIFT dataset, the running time for per query feature is about 3ms. If per given query image yields 1000 features, it can be classified or recognized in about 3s from an image dataset in the size of 1000.

## 6.5 Summary

In this chapter, we extended the use our context-sensitive framework on natural images. In the problem of tattoo segmentation, our first stage managed to segment skin regions, which provides spatial context information for tattoo segmentation. By splitting each tattoo image into clusters through a bottom-up process, we learned to merge the clusters containing skin and then distinguish tattoos from the other skin via top-down prior in the image itself. Tattoo segmentation with unknown number of parts is hence transferred to the second stage of our framework, a figure-ground segmentation concerning tattoo and skin. By running our algorithm on a tattoo database and evaluating the segmentation performance in terms of both accuracy and F-measure, the efficiency of our framework on tattoo segmentation was proved.

Beside directly evaluating the segmentation performance, we have also illustrated the benefit of tattoo segmentation on various tattoo classification tasks. By automatically segmenting the tattoo patterns, we managed to exclude noise from the background and proposed a novel tattoo classification system based on features from the tattoo patterns rather than the image. Also, our results show that our classifier based on shape-DNA outperforms the one based on SIFT. Therefore,

our system allows identification of the gang to which the tattoo at hand belongs and efficient large scale search.

# CHAPTER 7

# CONCLUSION

## 7.1 Summary of Contributions

### 7.1.1 Framework of Context-Sensitive Semantic Segmentation

We formulated the problem of context-sensitive semantic segmentation as a well-defined statistic model, proposed a two-stage framework, analyzed its efficiency, and showed how it can be applied to semantic segmentation tasks. Specifically, we developed a generalized context-sensitive framework that allows the use of not only appearance features but also context features. Through Bayes' theorem, we showed that the classical object-centered model is a special case of our context-sensitive model. The efficiency due to narrowing the searching space by context features is also explicitly explained. Further, we analyzed our framework in the form of the log probability, showed its relationship with the information theory and proved its efficiency in this view as well. By factorizing the context term using the chain rule in probability theory, we showed how our framework can be extensively used to employ varied context features and hence improve the performance of the context-sensitive semantic segmentation.

### 7.1.2 Context Object Segmentation in Nano Scale

In addition, we developed two algorithms of context object segmentation for spike segmentation. These algorithms first allow researchers to segment nano-scale membranes according to their closeness and varied profile shape, which is useful for many applications in visualization of plane-like structures in noisy data with high resolution. In the first algorithm, we developed the membrane segmentation on the local model of ridge-like membrane and thus it does not require the closed surface of the context object. We show its efficiency by applying on microvillus membrane segmentation. The use of receptive field model in the segmentation may also show its robustness in processing the noisy visual input. In the second algorithm, we developed the membrane segmentation on a global model based on level set function evolution. On one hand, it assumes the closed surface of the context object in each single slice along the axis that is parallel to the direc-

tion of the electron beam. On the other hand, it can tolerate the membrane with different profile shapes by considering membrane segmentation as a problem of object surface reconstruction. Such robustness derives from our hybrid model that combines appearance feature, the shape prior of the level set function and the localization prior propagated along the slices. The efficiency of our second membrane segmentation algorithm is demonstrated by applying on the task of HIV membrane outer surface reconstruction. Further, in related work on surface reconstruction, we developed an algorithm of reconstructing the semantic surface on 3D light microscopic images. By exploring the depth of field, we developed a prototype system that allows efficient acquisition of 3D drosophila reconstruction using a thin plate spline model. Therefore, our work provides the possibility to carry out context-sensitive semantic segmentation on not only electron microscopic images abut also light microscopic images.

### 7.1.3 Context-Sensitive Small Object Segmentation in Nano Scale

We also proposed context-sensitive small object segmentation on 3D data captured by nano-scale imaging. Our method is the first algorithm that allows automatic context-sensitive small object segmentation in nano-scale data and thus significantly reduces the workload of researchers in nano-scale visualization. The design and use of different context features help us reduce the search space of our target object accordingly. By incorporating them using our context-sensitive model in the second stage of our framework, nano-scale small object segmentation is efficiently achieved, where in comparison the state of the art semantic segmentation methods fail due to low SNR, low contrast, and high resolution. The excellent performance of our method was demonstrated on the tasks of microvillus and HIV spike segmentation. By comparing the time cost of the manual annotation and our algorithm on a tomogram with hundreds of thousands of spikes, we also demonstrated our algorithm can significantly accelerate the spike visualization procedure.

### 7.1.4 Context-Sensitive Tattoo Segmentation

We developed a context-sensitive semantic segmentation algorithm for extracting tattoos from images. Based on the idea of split-merge, our algorithm splits each tattoo image into clusters through a bottom-up process, learns to merge the clusters containing skin and then distinguishes tattoo from the other skin regions via top-down prior in the image itself. Tattoo segmentation with unknown number of clusters is thus transferred to a figure-ground segmentation. We applied

our context-sensitive segmentation algorithm on a single tattoo with different views and a tattoo dataset. The results demonstrated that our tattoo segmentation system is efficient. Based on the potential tattoo patterns provided by our tattoo segmentation, we further developed the first tattoo classification system based on tattoo patterns. We demonstrated state-of-the-art performance in various tasks of tattoo classification and also showed the superior of shape-DNA over SIFT in tattoo classification.

## 7.2 Future Work and Open Questions

### 7.2.1 Hierarchical Feature Space Exploration

In the sections 3 for context object segmentation, we manually select the scale where the context objects are salient for segmentation. According to the description of Section 2.4 of Chapter 2, the appearance features of the context objects in such scale presents relatively stronger bottom-up saliency in contrast to them in the other scales. One disadvantage of this strategy is that it may be tricky to manually tune the parameter $\sigma$ to produce the image with the appropriate scale for feature extraction. More importantly, manual scale selection is limited by the assumption that the context objects should be only salient in certain scales. Unlike electronic and light microscopic images, objects in some more general and natural images often consist of different parts that are salient in different scales. Thus it is insufficient to select a single scale where the object as whole appears to be salient. Thus there is a natural desire in having a hierarchical feature space where the feature responses from different scales that contribute to the saliency of the context object as whole are extracted for semantic segmentation.

### 7.2.2 Strategies in Context-Sensitive Semantic Segmentation

The method we have proposed is a step forward in nano-scale biological research to detect and visualize faint objects under extremely low SNR. However, there are still many open problems in this area. One of the major limitations of our method is its reliance on the performance of context object segmentation (membrane segmentation in the case of microvilli and HIV tomogram). Once a context object is missed, the related targets will be missed as well. On the other hand, context cues may be applied on the false-positive context objects and hence behave problematically. While we have developed means by which to control this trade-off (see semantic context cues in 5.3.3 for

details), this remains a serious limitation of our method. Indeed, one possible solution is to design context features based on context object likelihood, rather than the hard segmentation results.

Nano-scale context-sensitive semantic segmentation also presents problems in method evaluation. Most of the state-of-the-art methods on nano-scale semantic segmentation have always been plagued by the question of how to quantitatively evaluate the segmentation performance [108]. The gold standard is evaluation by annotations from experts on large datasets. Unfortunately, it is too expensive to fully annotate a nano-scale noisy dataset in practice. We avoided this problem by using data from a number of regions in the tomogram that are fully annotated by expert and thus were able to evaluate precisely which voxels were correctly identified. The existence of fully annotated dataset is indeed necessary for evaluation.

Last but not least, it is worth keeping in mind that the intrinsic limitation of electron tomography results in artifacts in nano-scale data and the segmentation results may not reflect the real nano-scale structure. Specifically, the limited tilt range produces a pair of regions with empty information in the Fourier space of the data. As a result, single-tilt axis leads to the missing wedge effect, which produces artifacts such as elongating, blurring or even fading the spatial features in real space. A detailed explanation of this effect and some possible solutions could be found in [60]. A model concerning the missing wedge effect should be included for the nano-scale segmentation model in the future.

### 7.2.3 Tattoo Segmentation

Similar to previous analysis that spike segmentation is sensitive to the membrane segmentation, tattoo segmentation and classification are also sensitive to skin segmentation. So far, our skin segmentation is based on the location prior that skin with tattoo more often appears in the center of the tattoo image in that tattoos are captured in purpose. This assumption limits the use of our methods in more general applications where the tattoos in the image is not captured in purpose (i.e.: surveillance) or does not appear as the most salient objects that draw our attention. To allow more general use of our tattoo segmentation and classification system, one way is to add the skin tone into skin segmentation. For the inhomogeneity due to shades on the skin, namely bias in the literature [78], an estimate of the bias field can be involved to overcome its influence. Another way is to involve mutual context information in an iterative manner. In our algorithm, the context object segmentation on skin narrows the searching space for the target object segmentation on tattoos.

Conversely, tattoo segmentation can also behave as context object segmentation to improve the target object segmentation on skin in fact. Thus it intuitively generates a loop between skin and tattoo segmentation that may iteratively improve the segmentation performance on both types of objects.

## 7.3 Closing Remarks

This thesis has explored the problem of semantic segmentation that is sensitive to context. As one solution for this problem, we have proposed a two-stage framework in which appearance features cooperate with varied context features to overcome the difficulties in semantic segmentation. By applying our framework on nano-scale spike segmentation and tattoo segmentation, we have demonstrated the efficiency of our novel framework in solving this challenging problem. Given that the quality of features substantially affects the performance of semantic segmentation, feature engineering is one of the most critical issues in semantic segmentation. Modeling and incorporating context features will allow more efficient processing and analysis on fundamental but sophisticated structures that are used to be done manually.

# APPENDIX A

# ANALYSIS ON THE PROBLEM OF OBJECT-CENTERED SEGMENTATION

Let the true labels of the noise and the target object be 1 and 2 respectively. Given a volume with extremely low SNR, for each of the N voxels in the volume, the appearance feature responses from object and noise are fairly close, which means $\Pr(f_i^A|o_i = 1) \approx \Pr(f_i^A|o_i = 2)$. Based on the Bayes' rule and Eq.(2.2), we can write the object likelihood function in Eq.(2.1) as:

$$\Pr(o_i|f_i) \simeq \frac{\Pr(f_i^A|o_i)}{\Pr(f_i^A)} \Pr(o_i). \tag{A.1}$$

Consequently, we have the discriminant function

$$\begin{aligned} g(f_i) &= \frac{\Pr(o_i = 1|f_i)}{\Pr(o_i = 2|f_i)} \\ &\simeq \frac{\Pr(f_i^A|o_i = 1)}{\Pr(f_i^A|o_i = 2)} \times \frac{\Pr(o_i = 1)}{\Pr(o_i = 2)} \end{aligned} \tag{A.2}$$

and use the following decision rule: labeled as the target object voxel if $g(f_i^A) > 1$; otherwise labeled as voxel of noise. The discriminant function is decomposed in two factors: the ratio of object likelihoods and priors respectively. Since the first factor is close to 1, the segmentation significantly depends on the ratio of the object and the noise priors, which is smaller than 1 because the SNR is quite low. Therefore, voxels are always classified as noise.

# APPENDIX B

# PROOF OF HYBRID SEMANTIC CONTEXT MODEL

*Proof.* As we assume the scale context feature $f_i^{C_{sc}}$ and the spatial context feature $f_i^{C_{sp}}$ are conditionally independent of each other given the semantic context feature $f_i^{C_{se}}$ and the target label $o'_i = 1$, we have

$$\begin{aligned}
&\Pr(o'_i = 1 | f_i^{A'}, f_i^{C_{sc}}, f_i^{C_{sp}}, f_i^{C_{se}}) \\
&\propto \Pr(o'_i = 1 | f_i^{A'}) \times \Pr(f_i^{C_{se}} | o'_i = 1) \times \Pr(f_i^{C_{sc}} | f_i^{C_{se}}, o'_i = 1) \times \Pr(f_i^{C_{sp}} | f_i^{C_{sc}}, f_i^{C_{se}}, o'_i = 1) \\
&= \Pr(o'_i = 1 | f_i^{A'}) \times \Pr(f_i^{C_{se}} | o'_i = 1) \times \Pr(f_i^{C_{sc}} | f_i^{C_{se}}, o'_i = 1) \times \Pr(f_i^{C_{sp}} | f_i^{C_{se}}, o'_i = 1)
\end{aligned}$$

According to Eq. (5.9),

$$\begin{aligned}
&\Pr(o'_i = 1 | f_i^{A'}, f_i^{C_{sc}}, f_i^{C_{sp}}, f_i^{C_{se}}) \\
&\propto \Pr(o'_i = 1 | f_i^{A'}) \times \Pr(f_i^{C_{se}} = 1 | o'_i = 1) \times \Pr(f_i^{C_{sc}} | f_i^{C_{se}} = 1, o'_i = 1) \times \Pr(f_i^{C_{sp}} | f_i^{C_{se}} = 1, o'_i = 1) \\
&\quad + \Pr(o'_i = 1 | f_i^{A'}) \times \Pr(f_i^{C_{se}} = 0 | o'_i = 1) \times \Pr(f_i^{C_{sc}} | f_i^{C_{se}} = 0, o'_i = 1) \times \Pr(f_i^{C_{sp}} | f_i^{C_{se}} = 0, o'_i = 1) \\
&= \Pr(o'_i = 1 | f_i^{A'}) \times \lambda \times \Pr(f_i^{C_{sc}} | f_i^{C_{se}} = 1, o'_i = 1) \times \Pr(f_i^{C_{sp}} | f_i^{C_{se}} = 1, o'_i = 1) \\
&\quad + \Pr(o'_i = 1 | f_i^{A'}) \times (1 - \lambda) \times \Pr(f_i^{C_{sc}} | f_i^{C_{se}} = 0, o'_i = 1) \times \Pr(f_i^{C_{sp}} | f_i^{C_{se}} = 0, o'_i = 1) \\
&= \lambda \Psi^C + (1 - \lambda) \Psi^A,
\end{aligned}$$

such that

$$\Psi^C = \Pr(o'_i = 1 | f_i^{A'}) \times \Pr(f_i^{C_{sc}} | f_i^{C_{se}} = 1, o'_i = 1) \times \Pr(f_i^{C_{sp}} | f_i^{C_{se}} = 1, o'_i = 1)$$

and

$$\Psi^A = \Pr(o'_i = 1 | f_i^{A'}) \times \Pr(f_i^{C_{sc}} | f_i^{C_{se}} = 0, o'_i = 1) \times \Pr(f_i^{C_{sp}} | f_i^{C_{se}} = 0, o'_i = 1).$$

Base on Eq. (5.3) and Eq. (5.5), we have

$$
\begin{aligned}
\Psi^A &= \Pr(o'_i = 1 | f_i^{A'}) \times \Pr(f_i^{C_{sc}} | f_i^{C_{se}} = 0, o'_i = 1) \times \Pr(f_i^{C_{sp}} | f_i^{C_{se}} = 0, o'_i = 1) \\
&= \Pr(o'_i = 1 | f_i^{A'}) \times \Pr(f_i^{C_{sc}} | f_i^{C_{se}} = 0, o'_i = 1) \times [\Pr(f_i^{C_{sp}} = 0 | f_i^{C_{se}} = 0, o'_i = 1) \\
&+ \Pr(f_i^{C_{sp}} = 1 | f_i^{C_{se}} = 0, o'_i = 1)] \\
&= \Pr(o'_i = 1 | f_i^{A'}) \times \Pr(f_i^{C_{sc}} | f_i^{C_{se}} = 0, o'_i = 1) \times [\Pr(f_i^{C_{sp}} > 1 | f_i^{C_{se}} = 0, o'_i = 1) \\
&+ \Pr(f_i^{C_{sp}} \leqslant 1 | f_i^{C_{se}} = 0, o'_i = 1)] \\
&= \Pr(o'_i = 0 | f_i^{A'}) \times \Pr(f_i^{C_{sc}} | f_i^{C_{se}} = 0, o'_i = 1) \\
&= [\Pr(f_i^{C_{sc}} = 1 | f_i^{C_{se}} = 0, o'_i = 1) + \Pr(f_i^{C_{sc}} = 0 | f_i^{C_{se}} = 0, o'_i = 1)] \times \Pr(o'_i = 1 | f_i^{A'}) \\
&= \Pr(o'_i = 1 | f_i^{A'}).
\end{aligned}
$$

□

# BIBLIOGRAPHY


[1] Scott T Acton and Adam Rossi. Matching and retrieval of tattoo images: Active contour cbir and glocal image features. In *Image Analysis and Interpretation, 2008. SSIAI 2008. IEEE Southwest Symposium on*, pages 21–24. IEEE, 2008.

[2] Shivani Agarwal and Dan Roth. Learning a sparse representation for object detection. In *Proceedings of the 7th European Conference on Computer Vision-Part IV*, ECCV '02, pages 113–130, 2002.

[3] Bogdan Alexe, Thomas Deselaers, and Vittorio Ferrari. Classcut for unsupervised class segmentation. In *Proceedings of the 11th European conference on Computer vision: Part V*, ECCV'10, pages 380–393, 2010.

[4] VI Arnold. Geometrical methods in the theory of ordinary differential equations (grundlehren der mathematischen wissenschaften). *Fundamental Principles of Mathematical Science.–Springer, Verlag, New York*, 250, 1983.

[5] David Arthur and Sergei Vassilvitskii. k-means++: the advantages of careful seeding. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, SODA '07, pages 1027–1035, Philadelphia, PA, USA, 2007. Society for Industrial and Applied Mathematics.

[6] Moshe Bar. Visual objects in context. *Nature Reviews Neuroscience*, 5(8):617–629, 2004.

[7] Jonathan T Barron, Mark D Biggin, Pablo Arbelaez, David W Knowles, Soile VE Keranen, and Jitendra Malik. Volumetric semantic segmentation using pyramid context features. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 3448–3455. IEEE, 2013.

[8] Alberto Bartesaghi, Guillermo Sapiro, and Sriram Subramaniam. An energy-based three-dimensional segmentation approach for the quantitative interpretation of electron tomograms. *Image Processing, IEEE Transactions on*, 14(9):1314–1323, 2005.

[9] G.C. Baylis and J. Driver. Shape-coding in it cells generalizes over contrast and mirror reversal, but not figure-ground reversal. *Nature Neuroscience*, 4(9):937–942, 2006.

[10] Jeffrey S Beis and David G Lowe. Shape indexing using approximate nearest-neighbour search in high-dimensional spaces. In *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, pages 1000–1006. IEEE, 1997.

[11] Serge Belongie, Jitendra Malik, and Jan Puzicha. Shape context: A new descriptor for shape matching and object recognition. In *NIPS*, volume 2, page 3, 2000.

[12] Serge Belongie, Jitendra Malik, and Jan Puzicha. Shape matching and object recognition using shape contexts. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(4):509–522, 2002.

[13] Elliot Joel Bernstein and Yali Amit. Part-based statistical models for object classification and detection. In *In IEEE Computer Vision and Pattern Recognition*, pages 734–740, 2005.

[14] Irving Biederman. Perceiving real-world scenes. *Science*, 177(4043):77–80, 1972.

[15] Irving Biederman, Robert J Mezzanotte, and Jan C Rabinowitz. Scene perception: Detecting and judging objects undergoing relational violations. *Cognitive psychology*, 14(2):143–177, 1982.

[16] Jose-Roman Bilbao-Castro, Carlos Oscar Sánchez Sorzano, Inmaculada García, and José-Jesús Fernández. Xmsf: Structure-preserving noise reduction and pre-segmentation in microscope tomography. *Bioinformatics*, 26(21):2786–2787, 2010.

[17] Andrew Blake and Michael Isard. *Active contours: the application of techniques from graphics, vision, control theory and statistics to visual tracking of shapes in motion*. Springer-Verlag New York, Inc., 1998.

[18] F.L. Bookstein. Principal warps: thin-plate splines and the decomposition of deformations. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 11(6):567 –585, jun 1989.

[19] Eran Borenstein, Eitan Sharon, and Shimon Ullman. Combining top-down and bottom-up segmentation. In *Proc. Computer Vision and Pattern Recognition Workshop Perceptual Organization in Computer Vision*, 2004.

[20] Eran Borenstein and Shimon Ullman. Class-specific, top-down segmentation. In *Proc. European Conf. Computer Vision*, volume 2, pages 109–124, 2002.

[21] Ali Borji and Laurent Itti. State-of-the-art in visual attention modeling. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(1):185–207, 2013.

[22] Y. Y. Boycov and M. P. Jolly. Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images. In *Eighth International Conference on Computer Vision*, pages 105–112, 2001.

[23] Yuri Boykov and Gareth Funka-Lea. Graph cuts and efficient nd image segmentation. *International Journal of Computer Vision*, 70(2):109–131, 2006.

[24] M.J. Brady and D. Kersten. Bootstrapped learning of novel objects. *Journal of Vision*, 3(6):413–422, 2003.

[25] John Canny. A computational approach to edge detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (6):679–698, 1986.

[26] Liangliang Cao and Feifei Li. Spatially coherent latent topic model for concurrent segmentation and classification of objects and scenes. *IEEE 11th International Conference on Computer Vision (2007)*, pages:1–8, 2007.

[27] Vicent Caselles, Francine Catté, Tomeu Coll, and Françoise Dibos. A geometric model for active contours in image processing. *Numerische mathematik*, 66(1):1–31, 1993.

[28] Vicent Caselles, Ron Kimmel, and Guillermo Sapiro. Geodesic active contours. *International journal of computer vision*, 22(1):61–79, 1997.

[29] Tony F Chan and Luminita A Vese. Active contours without edges. *Image processing, IEEE transactions on*, 10(2):266–277, 2001.

[30] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24(5):603–619, 2002.

[31] Daniel Cremers, Mikael Rousson, and Rachid Deriche. A review of statistical approaches to level set segmentation: Integrating color, texture, motion and shape. *Int. J. Comput. Vision*, 72:195–215, April 2007.

[32] Daniel Cremers, Nir Sochen, and Schn Christoph. A multiphase dynamic labeling model for variational recognition-driven image segmentation. *International Journal of Computer Vision*, 66:67–81, 2006.

[33] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893. IEEE, 2005.

[34] M. Demerec, editor. *Biology of Drosophila*. Hafner, New York, 1965.

[35] A. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society B*, 39(1):1–38, 1977.

[36] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification*. John Wiley and Sons, New York, second edition, 2001.

[37] Hossam E. Abd El Munim and Aly A. Farag. A shape-based segmentation approach: An improved technique using level sets. In *Proceedings of the Tenth IEEE International Conference on Computer Vision*, volume 2 of *ICCV '05*, pages 930–935, Washington, DC, USA, 2005. IEEE Computer Society.

[38] Boris Epshtein and Shimon Ullman. Semantic hierarchies for recognizing objects and parts. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 0:1–8, 2007.

[39] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.

[40] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Efficient graph-based image segmentation. *Int. J. Comput. Vision*, 59:167–181, September 2004.

[41] Wei Feng, Jiaya Jia, and Zhi-Qiang Liu. Self-validated labeling of markov random fields for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32:1871–1887, October 2010.

[42] Jose-Jesus Fernandez. Computational methods for electron tomography. *Micron*, 43(10):1010–1030, 2012.

[43] José-Jesús Fernández and Sam Li. An improved algorithm for anisotropic nonlinear diffusion for denoising cryo-tomograms. *Journal of structural biology*, 144(1):152–161, 2003.

[44] Jose-Jesus Fernandez and Sam Li. Anisotropic nonlinear filtering of cellular structures in cryoelectron tomography. *Computing in science & engineering*, 7(5):54–61, 2005.

[45] Daniel Freedman and Tao Zhang. Interactive graph cut based segmentation with shape priors. In *Proceedings of CVPR*, pages 755–762, 2005.

[46] Carolina Galleguillos and Serge Belongie. Context based object categorization: A critical survey. *Computer Vision and Image Understanding*, 114(6):712–722, 2010.

[47] Meirav Galun, Ronen Basri, and Achi Brandt. Multiscale edge detection and fiber enhancement using differences of oriented means. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE, 2007.

[48] GangInk. Gang tattoo databased, 2010. Available at `http://gangink.com/index.php?pr=GANG_LIST`.

[49] DM Greig, BT Porteous, and Allan H Seheult. Exact maximum a posteriori estimation for binary images. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 271–279, 1989.

[50] Lie Gu, Stan Z Li, and Hong-Jiang Zhang. Learning probabilistic distribution model for multi-view face detection. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 2, pages II–116. IEEE, 2001.

[51] Greg Hamerly and Charles Elkan. Alternatives to the k-means algorithm that find better clusterings. In *Proceedings of the eleventh international conference on Information and knowledge management*, CIKM '02, pages 600–607, New York, NY, USA, 2002. ACM.

[52] Hu Han and Anil K Jain. Tattoo based identification: Sketch to image matching. In *Biometrics (ICB), 2013 International Conference on*, pages 1–8. IEEE, 2013.

[53] Chris Harris and Mike Stephens. A combined corner and edge detector. In *Alvey vision conference*, volume 15, page 50. Manchester, UK, 1988.

[54] Junzhou Huang, Yunhong Wang, Tieniu Tan, and Jiali Cui. A new iris segmentation method for recognition. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, volume 3, pages 554–557. IEEE, 2004.

[55] Rui Huang, Vladimir Pavlovic, and Dimitris N. Metaxas. A graphical model framework for coupling mrfs and deformable models. In *Proceedings of CVPR*, pages 739–746, 2004.

[56] J. Hupe, A. James, B. Payne, S. Lomber, and J. Bullier. Cortical feedback improves discrimination between figure and background by v1, v2 and v3 neurons. *Nature*, 394:784–787, 1998.

[57] Anil K Jain, Jung-Eun Lee, and Rong Jin. Tattoo-id: Automatic tattoo image retrieval for suspect and victim identification. In *Advances in Multimedia Information Processing–PCM 2007*, pages 256–265. Springer, 2007.

[58] Anil K Jain, Jung-Eun Lee, Rong Jin, and Nicholas Gregg. Content-based image retrieval: An application to tattoo images. In *Image Processing (ICIP), 2009 16th IEEE International Conference on*, pages 2745–2748. IEEE, 2009.

[59] Wen Jiang, Matthew L Baker, Qiu Wu, Chandrajit Bajaj, and Wah Chiu. Applications of a bilateral denoising filter in biological electron microscopy. *Journal of structural biology*, 144(1):114–122, 2003.

[60] Hiroshi Jinnai and Richard J Spontak. Transmission electron microtomography in polymer research. *Polymer*, 50(5):1067–1087, 2009.

[61] C. Joao and S. Cristian. Constrained parametric min-cuts for automatic object segmentation. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[62] Michael J Jones and James M Rehg. Statistical color models with application to skin detection. *International Journal of Computer Vision*, 46(1):81–96, 2002.

[63] Michael Kass, Andrew Witkin, and Demitri Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 2:321–331, 1988.

[64] Leonard Kaufman and Peter Rousseeuw. *Clustering by means of medoids*. North-Holland, 1987.

[65] James R Kremer, David N Mastronarde, and J Richard McIntosh. Computer visualization of three-dimensional image data using imod. *Journal of structural biology*, 116(1):71–76, 1996.

[66] Frank R. Kschischang, Brendan J. Frey, and Hans-Andrea Loeliger. Factor graphs and the sum-product algorithm. *IEEEE Trans. on Information Theory*, 47(2):498–519, 2001.

[67] M.P. Kumar, P.H.S. Ton, and A. Zisserman. Obj cut. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 18–25, june 2005.

[68] V. Lamme. The neurophysiology of figure-ground segregation in primary visual cortex. *Journal of Neuroscience*, 15(2):1605–1615, 1995.

[69] Anne E Laumann and Amy J Derick. Tattoos and body piercings in the united states: a national data set. *Journal of the American Academy of Dermatology*, 55(3):413–421, 2006.

[70] Yvan G. Leclerc. Constructing simple stable descriptions for image partitioning. *International Journal of Computer Vision*, 3:73–102, 1989.

[71] Jung-Eun Lee, Rong Jin, Anil K Jain, and Wei Tong. Image retrieval in forensics: tattoo image database application. *MultiMedia, IEEE*, 19(1):40–49, 2012.

[72] Yong Jae Lee and Kristen Grauman. Collect-cut: Segmentation with top-down cues discovered in multi-object images. In *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, volume 0, pages 3185–3192, 2010.

[73] Bastian Leibe, Ales Leonardis, and Bernt Schiele. Combined object categorization and segmentation with an implicit shape model. In *In ECCV workshop on statistical learning in computer vision*, pages 17–32, 2004.

[74] Bastian Leibe, Aleš Leonardis, and Bernt Schiele. Robust object detection with interleaved categorization and segmentation. *International journal of computer vision*, 77(1-3):259–289, 2008.

[75] Bastian Leibe and Bernt Schiele. Interleaved object categorization and segmentation. In *In British Machine Vison Conference (BMVC'03)*, pages 759–768, 2003.

[76] Bastian Leibe and Bernt Schiele. Interleaving object categorization and segmentation. *Cognitive Vision Systems*, 3948:145–161, 2006.

[77] Anat Levin and Yair Weiss. Learning to combine bottom-up and top-down segmentation. *International Journal of Computer Vision*, 81:105–118, 2009.

[78] Chunming Li, John C Gore, and Christos Davatzikos. Multiplicative intrinsic component optimization (mico) for mri bias field estimation and tissue segmentation. *Magnetic resonance imaging*, 32(7):913–923, 2014.

[79] Chunming Li, Chenyang Xu, Changfeng Gui, and Martin D Fox. Level set evolution without re-initialization: a new variational formulation. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 430–436. IEEE, 2005.

[80] Tony Lindeberg. Scale-space for discrete signals. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 12(3):234–254, 1990.

[81] Tony Lindeberg. Edge detection and ridge detection with automatic scale selection. *International Journal of Computer Vision*, 30(2):117–156, 1998.

[82] G. Liu, Z. Lin, Y. Yu, and X. Tang. Unsupervised object segmentation with a hybrid graph model (hgm). *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 32(5):910–924, 2010.

[83] T. Liu, Z. Yuan, J. Sun, N. Zheng, X. Tang, and H. Y. Shum. Learning to detect a salient object. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[84] Xiuwen Liu, DL Wang, and Anuj Srivastava. Image segmentation using local spectral histograms. In *Image Processing, 2001. Proceedings. 2001 International Conference on*, volume 1, pages 70–73. IEEE, 2001.

[85] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[86] L Luccheseyz and SK Mitray. Color image segmentation: A state-of-the-art survey. *Proceedings of the Indian National Science Academy (INSA-A)*, 67(2):207–221, 2001.

[87] Vladan Lučić, Alexander Rigort, and Wolfgang Baumeister. Cryo-electron tomography: the challenge of doing structural biology in situ. *The Journal of cell biology*, 202(3):407–419, 2013.

[88] M. S. Boguski M. E. Fortini, M. P. Skupski and I. K. Hariharan. A survey of human disease gene counterparts in the drosophila genome. *Journal of Cell Biology*, 150:23–30, 2000.

[89] Y. Ma, H. Derksen, W. Hong, and J. Wright. Segmentation of multivariate mixed data via lossy data coding and compression. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 29(9):1–17, 2007.

[90] Yi Ma, Harm Derksen, Wei Hong, and John Wright. Segmentation of multivariate mixed data via lossy data coding and compression. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(9):1546–1562, 2007.

[91] Jakob H Macke, Nina Maack, Rocky Gupta, Winfried Denk, Bernhard Schölkopf, and Alexander Borst. Contour-propagation algorithms for semi-automated reconstruction of neural processes. *Journal of neuroscience methods*, 167(2):349–357, 2008.

[92] Ravikanth Malladi, James A. Sethian, and Baba C. Vemuri. Shape modeling with front propagation: A level set approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17:158–175, 1995.

[93] David R Martin, Charless C Fowlkes, and Jitendra Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(5):530–549, 2004.

[94] Antonio Martinez-Sanchez, Inmaculada Garcia, Shoh Asano, Vladan Lucic, and Jose-Jesus Fernandez. Robust membrane detection based on tensor voting for electron tomography. *Journal of structural biology*, 186(1):49–61, 2014.

[95] Antonio Martinez-Sanchez, Inmaculada Garcia, and Jose-Jesus Fernandez. A differential structure approach to membrane segmentation in electron tomography. *Journal of structural biology*, 175(3):372–383, 2011.

[96] Antonio Martinez-Sanchez, Inmaculada Garcia, and Jose-Jesus Fernandez. A ridge-based framework for segmentation of 3d electron microscopy datasets. *Journal of structural biology*, 181(1):61–70, 2013.

[97] Pierre Moreels and Pietro Perona. Evaluation of features detectors and descriptors based on 3d objects. *International Journal of Computer Vision*, 73(3):263–284, 2007.

[98] Greg Mori, Xiaofeng Ren, Alexei A. Efros, and Jitendra Malik. Recovering human body configurations: combining segmentation and recognition. In *Proceedings of the 2004 IEEE computer society conference on Computer vision and pattern recognition*, CVPR'04, pages 326–333, 2004.

[99] Eric N. Mortensen and William A. Barrett. Interactive segmentation with intelligent scissors. *Graph. Models Image Process.*, 60:349–384, September 1998.

[100] S Murugavalli and V Rajamani. An improved implementation of brain tumor detection using segmentation based on neuro fuzzy technique. *Journal of Computer Science*, 3(11):841–846, 2007.

[101] Amy Needham. Object recognition and object segregation in 4.5-month-old infants. *Journal of Experimental Child Psychology*, 78(1):3–22, 2001.

[102] Hieu Tat Nguyen, Marcel Worring, and Rein Van Den Boomgaard. Watersnakes: energy-driven watershed segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(3):330–342, 2003.

[103] The Federal Bureau of Investigation. The integrated automated fingerprint identification system.

[104] Stephen E Palmer. The effects of contextual scenes on the identification of objects. *Memory & Cognition*, 3:519–526, 1975.

[105] A. P. Pentland. A new sense for depth of field. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 9:523–531, 1987.

[106] Adrian Peter, Anand Rangarajan, and Jeffrey Ho. Shape lane rouge: Sliding wavelets for indexing and retrieval. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

[107] M. Peterson. Object recognition processes can and do operate before figure-ground organization. *Current Directions in Psychological Science*, 3:105–111, 1994.

[108] Grigore D Pintilie, Junjie Zhang, Thomas D Goddard, Wah Chiu, and David C Gossard. Quantitative analysis of cryo-em density map segmentation by watershed and scale-space filtering, and fitting of structures by alignment to regions. *Journal of structural biology*, 170(3):427–438, 2010.

[109] Paul C. Quinn and Philippe G. Schyns. What goes up may come down: perceptual process and knowledge access in the organization of complex visual patterns by young infants. *Cognitive Science*, 27(6):923–935, 2003.

[110] Xiaofeng Ren, Charless C. Fowlkes, and Jitendra Malik. Scale-invariant contour completion using conditional random fields. In *Proceedings of the 10'th IEEE International Conference on Computer Vision - Volume 2*, ICCV '05, pages 1214–1221, 2005.

[111] Xiaofeng Ren and Jitendra Malik. Learning a classification model for segmentation. In *In Proc. 9th Int. Conf. Computer Vision*, pages 10–17, 2003.

[112] Martin Reuter, Franz-Erich Wolter, and Niklas Peinecke. Laplace–beltrami spectra as shape-dnaof surfaces and solids. *Computer-Aided Design*, 38(4):342–366, 2006.

[113] Eraldo Ribeiro and Mubarak Shah. Computer vision for nanoscale imaging. *Machine Vision and Applications*, 17(3):147–162, 2006.

[114] C. Rother, V. Kolmogorov, and A. Blake. Grabcut–interactive foreground extraction using iterated graph cuts. In *Proc. ACM SIGGRAPH*, pages 309–314, 2004.

[115] Mirabela Rusu, Zbigniew Starosolski, Manuel Wahle, Alexander Rigort, and Willy Wriggers. Automated tracing of filaments in 3d electron tomography reconstructions using¡ i¿ sculptor¡/i¿ and¡ i¿ situs¡/i¿. *Journal of structural biology*, 178(2):121–128, 2012.

[116] Kristian Sandberg and Moorea Brega. Segmentation of thin structures in electron micrographs using orientation fields. *Journal of structural biology*, 157(2):403–415, 2007.

[117] Benjamin Schmid, Johannes Schindelin, Albert Cardona, Mark Longair, and Martin Heisenberg. A high-level 3d visualization api for java and imagej. *BMC bioinformatics*, 11(1):274, 2010.

[118] Yaar Schnitman, Yaron Caspi, Daniel Cohen-Or, and Dani Lischinski. Inducing semantic segmentation from an example. In *Computer Vision–ACCV 2006*, pages 373–384. Springer, 2006.

[119] Thomas Schoenemann and Daniel Cremers. Globally optimal image segmentation with an elastic shape prior. In *Computer Vision, IEEE International Conference on*, volume 0, pages 1–6. IEEE Computer Society, 2007.

[120] Eitan Sharon, Meirav Galun, Dahlia Sharon, Ronen Basri, and Achi Brandt. Hierarchy and adaptivity in segmenting visual scenes. *Nature*, 442(7104):810–813, 2006.

[121] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 8(22):888–905, 2000.

[122] Jamie Shotton, John Winn, Carsten Rother, and Antonio Criminisi. Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In *Computer Vision–ECCV 2006*, pages 1–15. Springer, 2006.

[123] Zheng Song, Qiang Chen, Zhongyang Huang, Yang Hua, and Shuicheng Yan. Contextualizing object detection and classification. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1585–1592. IEEE, 2011.

[124] H. Supper, H. Spekreijse, and V. Lamme. Contextual modulation in primary visual cortex as a neuronal substrate for working memory. *Journal of Vision*, 1(3):345, 2001.

[125] Antonio Torralba. Contextual priming for object detection. *International Journal of Computer Vision*, 53(2):169–191, 2003.

[126] Zhuowen Tu, Xiangrong Chen, Alan L. Yuille, and Song-Chun Zhu. Image parsing: Unifying segmentation, detection, and recognition. *International Journal of Computer Vision*, 63:113–140, 2005.

[127] Matthew A Turk and Alex P Pentland. Face recognition using eigenfaces. In *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on*, pages 586–591. IEEE, 1991.

[128] Luminita A. Vese and Tony F. Chan. A multiphase level set framework for image segmentation using the mumford and shah model. *Int. J. Comput. Vision*, 50:271–293, December 2002.

[129] Vladimir Vezhnevets, Vassili Sazonov, and Alla Andreeva. A survey on pixel-based skin color detection techniques. In *Proc. Graphicon*, volume 3, pages 85–92. Moscow, Russia, 2003.

[130] Luc Vincent and Pierre Soille. Watersheds in digital spaces: an efficient algorithm based on immersion simulations. *IEEE transactions on pattern analysis and machine intelligence*, 13(6):583–598, 1991.

[131] Niels Volkmann. A novel three-dimensional variant of the watershed transform for segmentation of electron density maps. *Journal of structural biology*, 138(1):123–129, 2002.

[132] G. Wahba, editor. *Spline Models for Observational Data*. SIAM, Philadelphia, PA, 1990.

[133] Britta Weber, Garrett Greenan, Steffen Prohaska, Daniel Baum, Hans-Christian Hege, Thomas Müller-Reichert, Anthony A Hyman, and Jean-Marc Verbavatz. Automated tracing of microtubules in electron tomograms of plastic embedded samples of¡ i¿ caenorhabditis elegans¡/i¿ embryos. *Journal of structural biology*, 178(2):129–138, 2012.

[134] Ross Whitaker, David Breen, Ken Museth, and Neha Soni. Segmentation of biological volume datasets using a level-set framework. In *Proceedings of the 2001 Eurographics conference on Volume Graphics*, pages 253–268. Eurographics Association, 2001.

[135] Christoph Winkler, Marlene Vinzenz, J Victor Small, and Christian Schmeiser. Actin filament tracking in electron tomograms of negatively stained lamellipodia using the localized radon transform. *Journal of structural biology*, 178(1):19–28, 2012.

[136] John Winn and Nebojsa Jojic. Locus: learning obeject classes with unsupervised segmentation. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 1, pages 756–763, 2005.

[137] John Winn and Jamie Shotton. The layout consistent random field for recognizing and segmenting partially occluded objects. In *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, volume 1, pages 37–44. IEEE Computer Society, 2006.

[138] Stella X. Yu and Jianbo Shi. Object-specific figure-ground segregation. In *In Proc. IEEE Comput. Soc. Conf. Comput. Vision and Pattern Recogn*, pages 39–45, 2003.

[139] Alan L. Yuille, Peter W. Hallinan, and David S. Cohen. Feature extraction from faces using deformable templates. *Int. J. Comput. Vision*, 8:99–111, August 1992.

[140] Lei Zhang and Qiang Ji. Image segmentation with a unified graphical model. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32:1406–1425, August 2010.

[141] Liang Zhao and Larry S. Davis. Closely coupled object detection and segmentation. In *Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1 - Volume 01*, pages 454–461, Washington, DC, USA, 2005. IEEE Computer Society.

[142] Ping Zhu, Jun Liu, Julian Bess, Elena Chertova, Jeffrey D Lifson, Henry Grisé, Gilad A Ofek, Kenneth A Taylor, and Kenneth H Roux. Distribution and three-dimensional structure of aids virus envelope spikes. *Nature*, 441(7095):847–852, 2006.

[143] Zhongjie Zhu, Yuer Wang, and Gangyi Jiang. Statistical image modeling for semantic segmentation. *Consumer Electronics, IEEE Transactions on*, 56(2):777–782, 2010.

[144] K. Zipser, V. Lamme, and P.H. Schiller. Contextual modulation in primary visual cortex. *Journal of Neuroscience*, 16:7376–7389, 1996.

# BIOGRAPHICAL SKETCH

Before joining the Florida State University to pursue his Ph.D, the author had obtained both his M.S. degree (2008) and B.E. degree (2005) of computer science from Sichuan University. He was a research intern at UtopiaCompression Corporation in Los Angeles, CA, USA (2012).