Image Segmentation for Object recognition & Combining AFM-CLSM for the study of Neuronal Growth Cones

by

Shripad Kondra

Thesis Supervisor: Prof. Vincent Torre

Submitted in total fulfillment of the requirements of the degree of Doctor of Philosophy

March 2009
Sector of Neurobiology
Scuola Internazionale Superiore di Studi Avanzati

To Mom & Dad

Abstract

The Marie curie research training project aims training to the researcher, and to apply the learnt knowledge. During the initial phase computer vision methods are learnt, and in the later part this knowledge was applied to solve problems in neuroscience. The work is primarily divided into 3 parts. The major part of thesis was to implement a general segmentation algorithm which can be used to extract the object in a bottom-up approach. An attempt was made to introduce new texture feature. Finally the gained knowledge was applied to creating an application for neuronal growth cone analysis by combining multi-modal information from Atomic force microscopy and confocal imaging.

Image segmentation has been a classical and still unsolved problem in Computer vision, Image containing texture makes the problem more difficult. Segmentation is also one of the first steps in many image analysis tasks and many visual cues, such as color, texture and illumination help in achieving segmentation. Segmentation is also goal dependent, subjective, and hence ill-posed in a general set up. However, it is desirable to consider generic criteria that can be applied to a large variety of images and can be adapted for specific applications. The first part of the thesis proposes a method for image segmentation based on a watershed algorithm that uses edges (colour+intensity+scale) and a novel region-merging procedure. The merging procedure uses a $\chi 2$ distance over three feature vectors: one for color representation, one for the scale map, and one describing the output of rotational filters. Visual representation of texture, similar to a Texton map is obtained with a modified Lindeberg scale. This scale map is also used for eliminating edges in the region of fine scales. Performance of the method is evaluated on images from the Berkeley database. We also collected a mammal database containing 6 classes. The classification is done using the shape extracted from the segmentation by k-NN classifier.

The second part of the thesis focuses on texture classification. Our method uses only 3 circular filters. Images are first filtered using these filters, then thresholded and averaged over two small neighborhoods. Universal textons are generated without learning from the training sets. 80 universal textons are used for each neighborhood. The feature space is reduced in one neighborhood by grouping into 4 bins. Each image is thus represented by a 2D histogram giving a 320 (80 x 4) dimensional feature vector (Model). Models are then trained with Support Vector Machines using $\chi 2$ kernel. The results are compared with state of art texture classification methods on 4 texture databases. The proposed method is best on one of this database. Further this texture feature is combined in a form of texton map and is used as texture information in the above proposed segmentation method

In the third part we present a simple method to combine Atomic force microscopy (AFM) and confocal images to study differentiating embryonic stem (ES) cells-derived and dorsal root ganglia (DRG) neurons in culture. AFM provides the possibility to map the 3D structure of viewed objects with a nanometric resolution, which cannot be achieved with other imaging methods such as conventional video imaging and confocal fluorescent microscopy. Confocal imaging allows the simultaneous acquisition of immunofluorescence images. In order to reduce changes of the image surface caused by the pressure of the AFM tip, we have developed a procedure to obtain 0 pN scanning force images. We also observed topographical structures with nanoscale dimensions which are referred to as “invaginations” or “holes”. Such studies can lead to a better understanding of neurodegenerative diseases where the mechanical and morphological properties of neurons and of growth cones in particular can be of great importance.

Acknowledgements

First and foremost I want to express my gratitude and thanks to my thesis advisor, Vincent Torre. I feel Vincent has been an ideal adviser. He has been a great mentor, a collaborator, and a guide. This thesis would not have been completed without his commitment and diligent efforts which not only influenced the content of the thesis but also the language in which it has been conveyed. He not only showed how to exhaustively explore different methodologies and analyze results, but also gave advice to me regarding the need for searching the answer to why things were not working instead of choosing another solution which was a great piece of advice I will always cherish. I am also grateful to him to make the PhD research more applicable and useful in the field of neuroscience. I thank him for offering me the opportunity to do a PhD on such interesting and challenging topic and providing me the platform for this thesis. I can never forget the support Vincent showed towards the fag end of my thesis when he learned about my job offer and the constraint of finishing the remaining part of research and the dissertation in two months time. I would like to thank Alessandro Verri, my thesis examiner, for his interest in my work.

I am grateful to Ulrich Hofmann, my MS supervisor for introducing me to the field of neuroscience. I am also grateful to Radu Horaud, the VISIONTRAIN project coordinator. Radu has always been very open and a great support. I would like to thanks Michael Lindenbaum for hosting me during my Internship in Haifa, from whose association I have tremendously benefited. I would also like to give special thanks to Andrew Zisserman, whose discussions and motivation helped me to do work on texture classification.

I would like to thank all my friends and colleagues from Neurobiology sector for making SISSA a fun and interesting place to work. Much respect to my office mates Walter, Reza, Jummi, Rajesh, Diana, Anil for putting up with me during these past three years. I shared many interesting discussions and work time fun with them. The help received from them both personally and professionally is also immense, and Jummi's aid has not stopped pouring even after my shift from Trieste. In the last year of my PhD, the differential discussions (Dr. House rules) with Diana, Jummi and Rajesh helped solve difficult problems quickly both mine and theirs. I also want to thank Elisabetta, Jelena, Giacomo, Jacobo, Fred, Paolo, Valentina, Elisa, and Anu for the lively working atmosphere. I would like to thank the collaborators of VISIONTRAIN for providing me an excellent platform to learn different aspects of computer vision. Special thanks to Ketut and Gal for giving me advice whenever was needed. I would also like to thank Alex Meehan, Manuela Lough, Amanda Colombo, Maria Sole and Riccardo Iancer for bearing with me for all the administrative goof ups.

I can never forget my good friends Pierdavide, Jamie, Carlo, Fabio, Davide, Fabrizio, Stefano, Sajid, Nicola, Carina, Petra and Niels for making my stay at Trieste so much more pleasant. They introduced me to the Italian culture and language. They were like a second family to me and were always ready to help and guide me. I still can not help missing delicious Tiramisu that Fabrizio used to make. I also cannot forget the Starcraft wars played on weekend with Jamie and Fabio. Many thanks to my Yoga guru Francesco; it was really special to learn from him the neuroscience view of yoga.

I cannot end without thanking my parents, for their absolute confidence in me. My final words go to Rucha, my to-be wife. In my work she suffered the most. Thank you for putting up with my late hours, many spoiled weekends and above all for staying by my side.

“Tu e me insieme niente siete impossibile nel mondo”

Contents

1 Introduction

Figure 1.1 What is an Object?

Ever since computer vision and machine vision field evolved, there was a parallel field trying to solve the vision problem through psychophysical point of view. Many recent algorithms are motivated by how humans might be solving it. The most famous example is the work done by Nobel laureates Hubel and Wiesel; their work established a foundation for visual neurophysiology, describing how signals from the eye are processed by the brain to generate edge detectors, motion detectors, stereoscopic depth detectors and color detectors, building blocks of the visual scene. Scale space theory led to creation of the most cited and widely used descriptors the SIFT. Textons introduced by [1;2] all are motivated from neuroscience. Ever since the Berkeley database [3] with human segmentations was published, there has been a competition and say a target to make machine algorithms like edge detection and segmentation work as humans do. Benchmarking algorithms has become more popular. It is not far that we will reach the human goal. The first question which needs attention, is to understand what is the meaning of an object? Given a scene, humans can easily categorize an image. However they can also modify their categorization given top level information, for example to search for a particular object inside a scene, something which is still a challenge for machine vision. Figure 1.1 shows an example scene. Segmentation criteria can be from more specific (just a house and a background) to more general (house, grass, tree and sky), or a more search oriented to find a window in the scene. Classification is useful because objects within the same category share so many properties and behaviors. All chairs are not exactly alike, nor are all hamburgers, but one chair is a lot more like another than it is like any hamburger, and vice versa [4]. Previous experience with members of a given category therefore allows us to predict with reasonable certainty what new members of that same class will do. As a consequence, we can deal with most new objects at the more abstract level of their category, even though we have never seen that particular object before. Classifying objects as members of known categories is actually very difficult. Consider the wide variety of cats shown in, for example. How can we recognize almost immediately that they are all cats? Do cats have some unique set of properties that enable us to perceive them as cats? What can these unique features be? These problems are one of the most difficult and as yet unsolved puzzles of visual perception.

Figure 1.2 Variation in the category of cats; which feature do they share in common?

1.1 The Goal

The thesis targets three main problems. First it addresses the issue of image segmentation in particular segmentation of natural images and how a shape can be extracted in an unsupervised way without the previous knowledge of object category. Second goal was to introduce new texture representation to improve segmentation, which was later specified to obtain a fast and robust texture classification method using a bank of filters. The third goal was to study the morphological characteristics of neuronal growth cones using multimodal methods like atomic force microscopy and confocal imaging.

1.1.1 Image Segmentation for Object Recognition

The main target of the thesis is to understand how humans perform segmentation and implement a robust segmentation procedure. Texture segmentation is further a difficult problem. A Textured region can contain texture elements of various sizes, each of which can itself be textured. The segmentation method combines color, intensity and texture. Figure 1.3 shows a pictorial representation of the research problem. The wider aim is to use this segmentation to extract or label object category in the image. The problem becomes more difficult when the categorization is done in the same class. For this we collected a database of six classes of mammals.

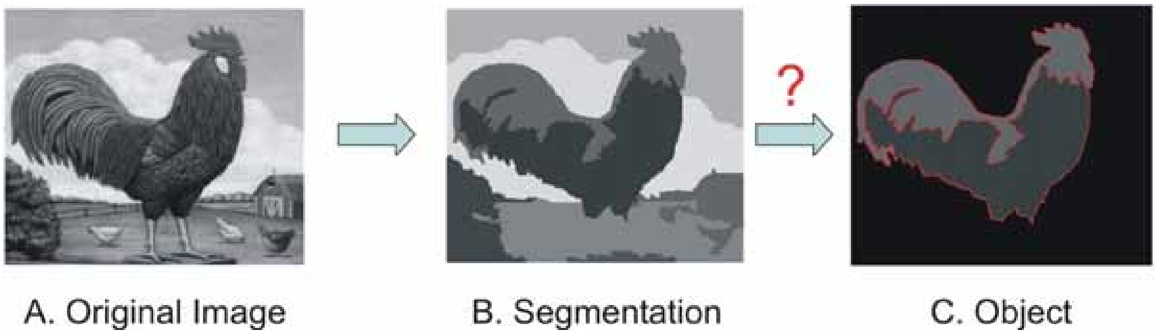

A. Original Image
B. Segmentation
C. Object

Figure 1.3 Research Problem

1.1.2 Texture Classification

Within this thesis the main problem addressed is the classification of textures. Classification is one of the main areas of current texture research, other major areas being segmentation, synthesis and shape from texture. The general texture classification problem can be defined as follows; a texture model of some description is generated from a set of training images of which the texture classes are known. We use a filter based approach, by generating textons and representing image with texton maps and then constructing a 2d histogram from this maps to form the feature model. The system is then presented with a novel image containing textural information; it is then to correctly identify the class of this novel texture. This is in contrast to segmentation where the aim is to divide the image up into regions of separate textures but the actual texture class itself is unknown, all that is required is to divide the image into textured regions.

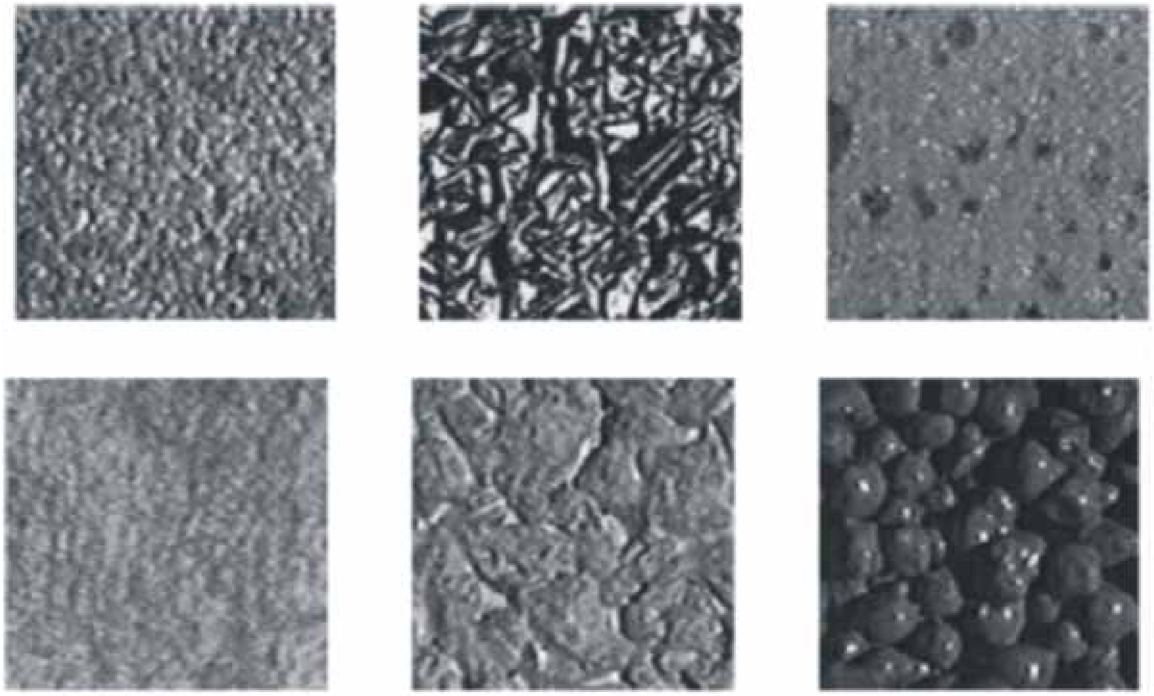


Figure 1.4 How is texture represented?

1.1.3 Neuronal Growth Cone Analysis

A **growth cone** is a dynamic, actin-supported extension of a developing axon seeking its synaptic target. Their existence was originally proposed by Spanish histologist Santiago Ramón y Cajal based upon stationary images he observed under the microscope. Neuronal growth cones are situated on the very tips of nerve cells on structures called axons and dendrites. The sensory, motor, integrative, and adaptive functions of growing axons and dendrites are all contained within this specialized structure.

The morphology of the growth cone can be easily described by using the hand as an analogy. The fine extensions of the growth cone are known as "filopodia" or microspikes. The filopodia are like the "fingers" of the growth cone; they contain bundles of actin filaments (F-actin) that give them shape and support. Filopodia are the dominant structures in growth cones, and they appear as narrow cylindrical extensions which can extend several microns beyond the edge of the growth cone. The filopodia are bound by membrane which contains receptors and cell adhesion molecules that are important for axon growth and guidance.

In between filopodia--much like the webbing of the hands--are the "lamellipodia". These are flat regions of dense actin meshwork instead of bundled F-actin as in filopodia. They often appear adjacent to the leading edge of the growth cone and are positioned between two filopodia, giving them a “veil-like” appearance. In growth cones, new filopodia usually emerge from these inter-filopodial veils.

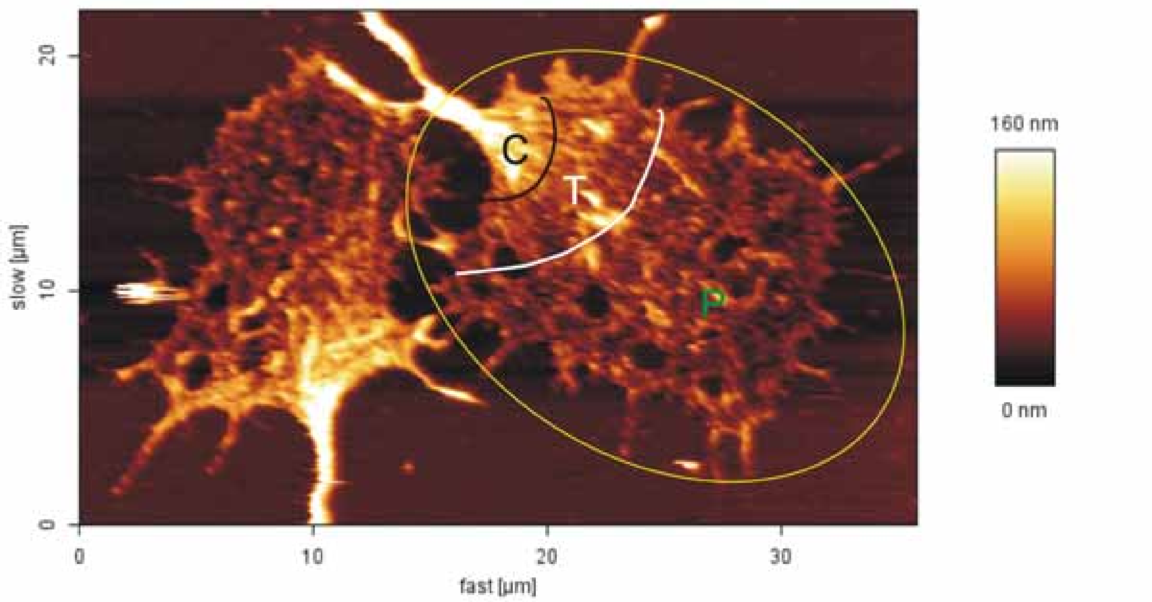


Figure 1.5 Atomic Force Microscopy Image of a Neuronal Growth cone

The growth cone is described in terms of three regions: the peripheral (P) domain, the transitional (T) domain, and the central (C) domain as shown in Figure 1.5. The yellow ellipse marked is the entire growth cone. The peripheral domain is the thin region surrounding the outer edge of the growth cone. It is composed primarily of an actin-based cytoskeleton, and contains the lamellipodia and filopodia which are highly dynamic. Microtubules, however, are known to transiently enter the peripheral region via a process called dynamic instability. The central domain is located in the center of the growth cone nearest to the axon. This region is composed primarily of a microtubule-based cytoskeleton, is generally thicker, and contains many organelles and vesicles of various sizes. The transitional domain is the region located in the thin band between the central and peripheral domains.

Goal in this thesis is to study and characterize the formation of holes in the neuronal growth cones. The main problems which needed to be addressed were as follows

- The images acquired using the two modalities were of different resolution, hence needed to be registered.
- The images acquired using AFM depends on the force used for scanning. So one of the problems was how to make it independent of this criterion.
- It was also necessary to verify if the analysis was dependent on the method used for fixation of growth cones.

1.2 Outline of Dissertation

This chapter introduces the research problems and presented the overall goal of the thesis. It then continued with a brief background on texture classification and outlined our approach to the problem. Next it also introduced the problems of biological application which has been addressed in this thesis.

Chapter 2 describes our new segmentation method. It starts by mentioning some of the recent state of art methods explaining, the block diagram of the proposed segmentation scheme. It briefly describes the edge detection method used. Scale edges have been introduced in this chapter which is combined with gradient and color edges. An initial segmentation is obtained using these edges. A Link method for merging the segments is explained. The chapter ends with the results obtained using our segmentation method.

Chapter 3 compares our segmentation with other state of art methods for which the results on the Berkeley datasets have already been published. It mentions the benchmark strategy and the Berkeley dataset for human segmentations in brief.

Chapter 4 mentions about the mammal dataset we collected. We describe how we extracted the shapes by using our segmentation method. Using a shape-based recognition scheme we do the classification on this dataset. Comparison of our results with the widely used "bag of feature" method is described at the end.

Chapter 5 introduces our new feature for texture classification using only three circular filters. It mentions about our new concept of universal textons. Further a comparison of our method is done with state of art methods on four challenging texture datasets.

Chapter 6 describes how we combined multi-modal information viz. Atomic force microscopy and Confocal imaging to study neuronal growth cones. Two registration methods are implemented and discussed. A new concept of zero pN scanning force is introduced. The chapter ends by describing the morphological analysis of growth cones.

Chapter 7 summarizes the thesis and the key contributions. It further the mention the limitations of the segmentation and texture feature used and proposes a probable alternative solution and some advances done in these fields during the writing of this thesis. It also provides some suggested directions for future research in these areas.

“Quando si chiude una porta si apre un portone”

2 Image Segmentation

It is necessary to partition the image in some way and separate the background where the objects stand from. This step, which is between low and high-level vision is usually referred as image segmentation or partition [5]. While it has been clear since the time of the Gestalt movement in psychology [6] that image-based segmentation and grouping play a powerful role in human visual perception, it has proven difficult to specify precise segmentation criteria and to develop efficient algorithms for computing segmentations that capture non-local properties of an image. A wide range of computational vision problems could in principle make good use of segmented images, were such segmentations are reliably and efficiently computable. For instance intermediate-level vision problems such as stereo and motion estimation require an appropriate region of support for correspondence operations. Spatially non-uniform regions of support can be identified using segmentation techniques. Higher-level problems such as recognition and image indexing can also make use of segmentation results in matching, to address problems such as figure-ground separation and recognition by parts. There is a large literature on segmentation and clustering, with applications in many areas other than computer vision. [7]

2.1 Challenges

Camouflage is striking evidence that segmentation can be hard not only for computers [8]. Animal fur, painted carefully to match the color and texture of its habitat, provides some textures cannot be characterized by a single scale, such as when a texture is composed of several elements that are differently spaced, or when the elements themselves are textured, a hiding place for a predator and a safe refuge for a prey. Computer systems often have difficulties to properly segment even simpler images. In particular these systems face difficulties when the image includes complex textures. Illumination varies considerably, for example direct sunlight and shadows during the day to artificial or dim lighting at night. Although models of color and illumination invariance have made significant advances, they still are far from being effective solutions when compared to human and mammalian visual systems which are extremely well adapted to such changes. A robust segmentation algorithm must handle color changes and provide invariance to a broad range of illumination and lighting changes.

2.2 State of Art

There are many papers dealing with automatic segmentation. A universal algorithm for segmenting images does not exist and, most techniques are tailored on particular applications and may work only under certain hypotheses. A classification of segmentation algorithms is not always straight-forward since some techniques resort to more than one strategy to achieve segmentation and thus cannot be sharply categorized. The most widely used approaches are clustering based approaches, split and merge, region growing techniques, edge-based, and neural network based. A more detailed review of segmentation methods can be found in [9].

We have to mention the well known work [10] based on normalized cuts which segments an image into non-overlapping regions. They introduced a modification of graph cuts, namely normalized graph cuts, and provided an approximate closed-form solution. They represent the set of points in an arbitrary feature space as a weighted undirected graph G = (V ;E), where the nodes are the points in the chosen feature space and edges are established between each pair of nodes. The weight w(i; j) of each edge is a function of the similarity between nodes i and j. The goal is to partition the set of vertices V into disjoint sets V_1...... V_M such that a predefined similarity measure is high for vertices within the same set and low across different sets. However, the boundaries of detected regions often do not follow the true boundaries of the objects. An improved approach using Affinity functions of patch-based (using CIELAB histogram and the χ^2 distance) and gradient-based distance (Combining brightness, color and texture) were used by [11] to obtain an automatic segmentation which was fed to an eigensolver to cluster the image. Our approach is close to [11] with respect of combining gradient and histogram features however we don't use the complex clustering approach but use simple region merging. In our case gradient information is used to obtain the initial segmentation while histogram representation is used for merging. There is another direction in image segmentation by using Level Set Methods [12]. The boundary of a textured foreground object is obtained by minimization (through the evolution of the region contour) of energies inside and outside the region. A versatile segmentation procedure combining edges, scale map and watershed segmentation has been proposed by [13]. Original images are first regularized by using a self-adaptive implementation of the Mumford-Shah functional so that the two parameters and controlling the smoothness and fidelity, automatically adapt to the local scale and contrast of the original image. From the regularized image which is piecewise smooth, a piecewise constant image is obtained representing a segmentation of the original image. Our approach uses the modified scale map from [13] but does not use the regularization step. [14] have proposed a method that automatically partitions a single image into non-overlapping regions coherent in texture and color. An assumption that each textured or colored region can be represented by a small template, called the seed, is used. Positioning of the seed across the input image gives many possible sub-segmentations of the image having same texture and color property as the pixels behind the seed. A probability map constructed during the subsegmentations helps to assign each pixel to just one most probable region and produce the final pyramid representing various detailed segmentations at each level. Each sub-segmentation is obtained as the min-cut/max-flow in the graph built from the image and the seed. Our approach uses the illumination invariant merging procedure from their approach.

We would like to mention a method [15] which was published recently, they formulate single-image multi-label segmentation into regions coherent in texture and color as a MAX-SUM problem for which efficient linear programming based solvers have recently appeared. There exists another approach for image segmentation using local variation [16]. An important characteristic of their approach is that it is able to preserve detail in low-variability regions while ignoring detail in high-variability regions. The algorithm proposed by them is very fast and also gives excellent results but is vulnerable to parameter settings which can vary from image to image. Another approach [17] determines all salient regions of an image and builds them into a hierarchical structure and the segmentation is done by weighted aggregation and consists of fine-to-coarse pixel aggregation. Aggregates of various sizes, which may or may not overlap, are revealed as salient, without predetermining their number or scale. Our method is close to this approach in the sense that we also use an aggregation based merging step to merge many segments at once. Another method [18] uses image segmentation technique using the morphological Waterfall algorithm. Joint detection and segmentation of a single object class has been achieved by several authors [19;20]. Typically, these approaches exploit a global shape model and are therefore unable to cope with arbitrary viewpoints or severe occlusion. Additionally, only highly structured object classes are addressed. [21] have investigates the problem of achieving automatic detection, recognition and segmentation of object classes.

While there are other approaches to image segmentation that are highly efficient, these methods generally fail to capture perceptually important non-local properties of an image.

2.3 Block Diagram

Boundaries and regions are closely intertwined [11]. A closed boundary generates a region while every image region has a boundary. Psychophysics experiments suggest that humans use both boundary and region cues to perform segmentation [22]. In order to build a vision system capable of parsing natural images into coherent units corresponding to surfaces and objects, it is clearly desirable to make global use of both boundary and region information.

The block diagram of our segmentation scheme is shown in Figure 2.1. Edge detection is another wide area of research; Even state-of-art edge detectors are not able to define closed boundaries, while some of them are too slow for real world applications. We use an efficient biologically motivated edge detection method [Pellegrino et al] which combines multi-scale and multi-directional filtering. Watershed algorithm is used on the distance transform of these edges to complete the broken edges and thus obtaining an initial over-segmentation. The segments are further merged using a novel Link method.

Figure 2.1 Segmentation scheme

2.4 Normalization

Images can be of different resolution. In order to reduce the complexity of determining parameters and the scales of edge detector for each and every image, we resize every image using bicubic interpolation such that the size of diagonal is 380 pixels. The scales used for edge detection thus remain unchanged for the segmentations shown in the thesis. We adopt the edge detection strategy as in [23]. The edge detection combines an extensive spatial filtering with classical methods of computer vision and newly developed algorithms. Step edges are computed by extracting local maxima from the energy summed over a large bank of directional odd filters with a different scale. We use three high scales so that we don't capture fine details.

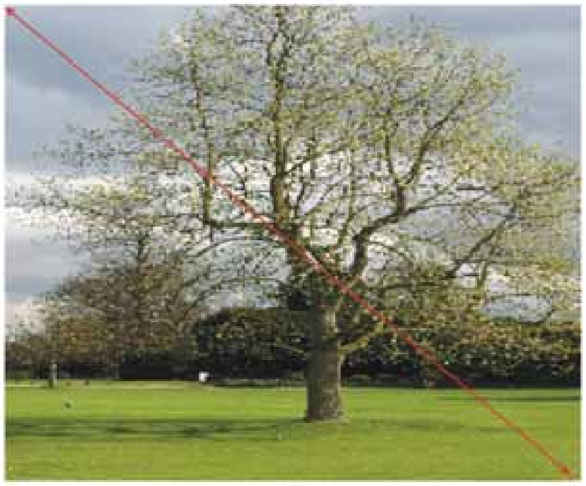


Figure 2.2 Image normalization:
image is resized using bicubic interpolation so that diagonal is 380 pixels

2.5 Scale Map as Texture Information

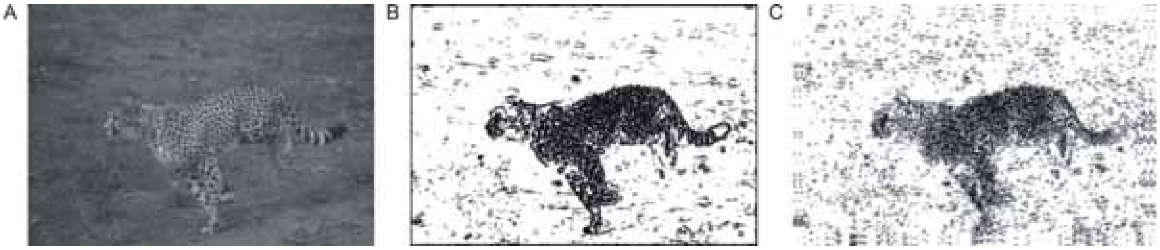


Figure 2.3 Image containing texture, its scale map & rotational scale map

We refer to the scale as $2\pi\sigma$, where σ is the standard deviation of the Gaussian filter. As suggested by [24;25], the local scale can be estimated considering the normalized derivatives

$$G_\lambda = t^{\frac{\lambda}{2}}\sqrt{{L_x}^2 + {L_y}^2} \qquad \textbf{eq. 1}$$

Where L_x and L_y are the x and y derivatives of the original image convolved with the Gaussian filter $\exp(-(x^2 + y^2)/2t)$ with $t = \sigma^2$. These normalized derivatives $G_\lambda(t)$ depend on the value of the parameter λ . As discussed by [24;26], the most convenient choice for the Gaussian step edges is $\lambda = \frac{1}{2}$. A local maximum of $G_{\frac{1}{2}}(t_{\max})$ for $t = t_{\max}$ indicated the presence of feature having the $scale = 2\pi\sqrt{t_{\max}}$ provided that $G_{\frac{1}{2}}(t_{\max})$ is larger than 1.85 and larger than 1 percent of the maximum of the other values of $G_{\frac{1}{2}}(t)$. In this thesis we modify eq. 1 mention above to include all the derivatives in 8 directions. We analyze the image with four small scales [3, 5, 7, 9] as texture is nothing but fine details. In similar line to extract the rotational information from the image we use Laplacian of Gaussian filters at 6 different scales. The modified scale map and rotational map are shown in Figure 2.3. As can be noted, the cheetah pops out of the background in the scale map. We use this information to extract scale edges as described in [27] In brief a gradient-based paradigm is used. At a location (x,y) in the image, a circle is drawn of radius r, and is divided along the diameter at orientation θ . The gradient function $G(x, y, \theta, r)$ compares the contents of the two resulting disc halves. A large difference between the disc halves indicates a discontinuity in the image along the disc's diameter. The half-disc regions are described by histograms of the four scales from the scale map, which are compared with the χ^2 histogram difference operator. The resulting scale edges for different images can be seen from Figure 2.4. It should be noted that this edges are not able to distinguish among fine textures with different properties, and just aim at detecting the presence of textures.

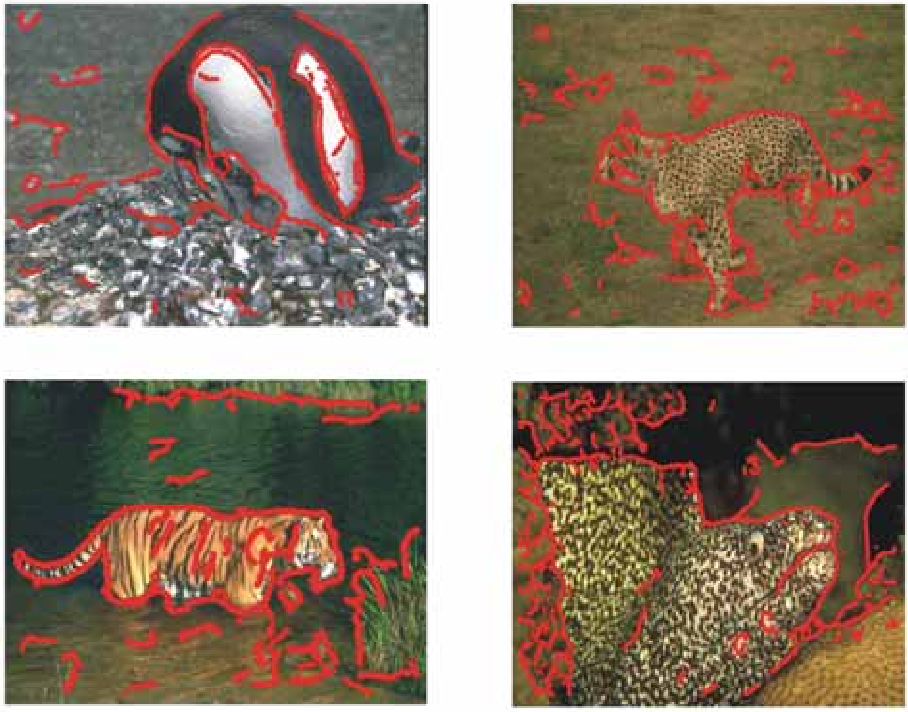
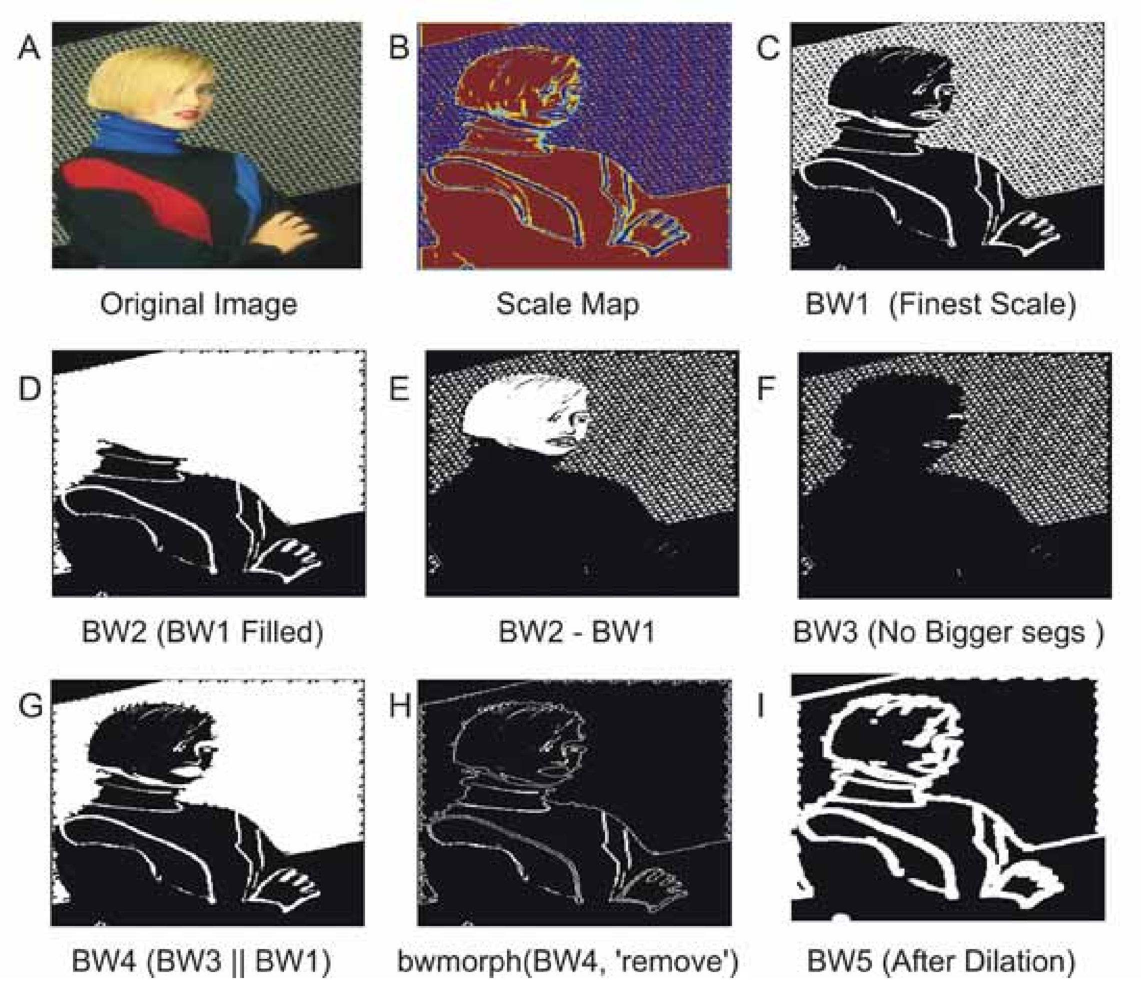


Figure 2.4 Scale edges detected from the scale map

2.6 Combined Edge Detection

Figure 2.5 Comparison of old and new approach for Edge detection

The problem with edge detection approach used was that if a low threshold was applied to the gradient function it detected fine texture edges which resulted in an over segmentation in the range greater than 1000 segments. If high threshold was used not all edges are detected. To overcome this approach we make efficient use of the scale map detected in the previous section. The idea is to morphologically segment the fine scale region and suppress the edge information in one of the levels (scales) used for edge detection. Figure 2.6.I shows the region where fine detailed information should be present. Figure 2.5 shows the result before and after the application of this step.



Original Image | Scale Map | BW1 (Finest Scale)

BW2 (BW1 Filled) | BW2 - BW1 | BW3 (No Bigger segs)

BW4 (BW3 || BW1) | bwmorph(BW4, 'remove') | BW5 (After Dilation)

Figure 2.6 Using scale map to suppress fine texture edges

The main idea of our approach was to start with strong edges. As shown in Figure 2.7 edges are detected separately for brightness gradient and color gradients, brightness gradient is calculated on L channel of CIELAB space. While for color we use two channels “Green-Blue” and “Green-Red”. Edges are combined with a simple or operation. Finally scale edges are added. So the final edge map is the combination of Gradient, Color and Scale edges. The notable advantage of using this scale edges can be seen from Figure 2.7C where the Gradient and color edges failed to detect the boundary between the tree and the cloud.

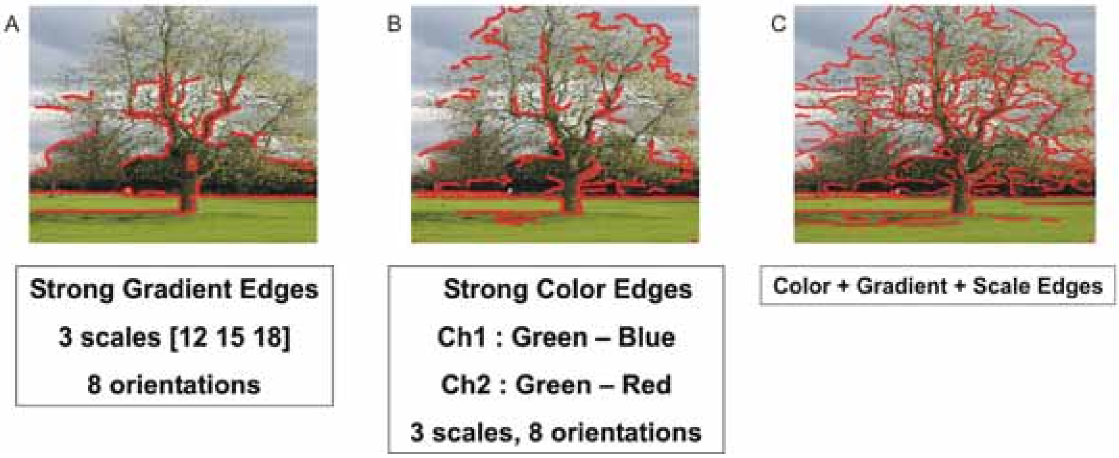


Figure 2.7 Combined Edge detection

2.7 Watershed Segmentation

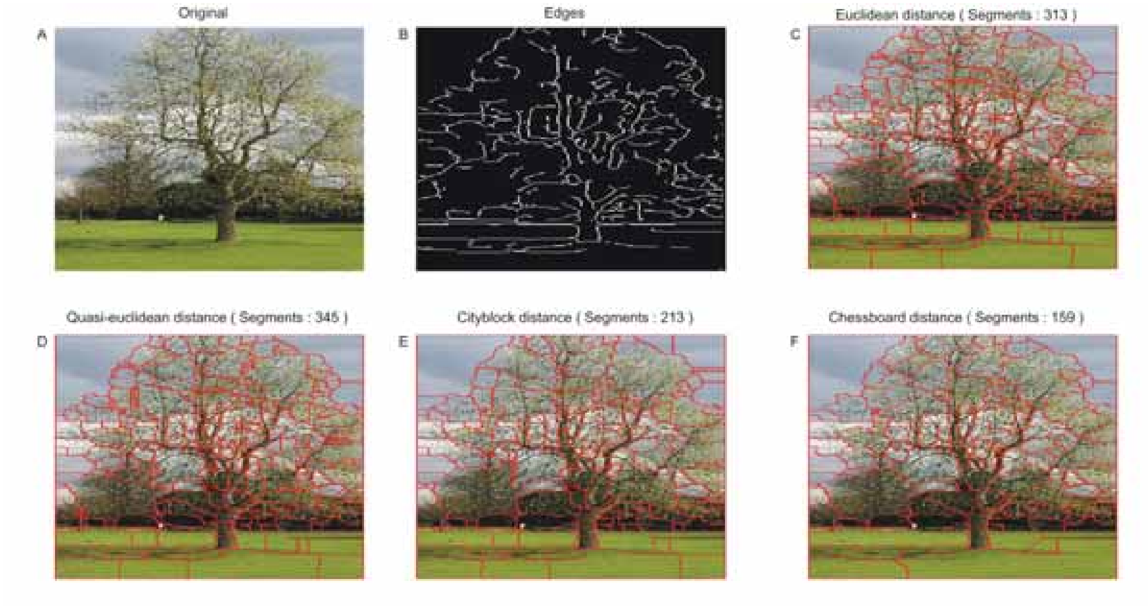
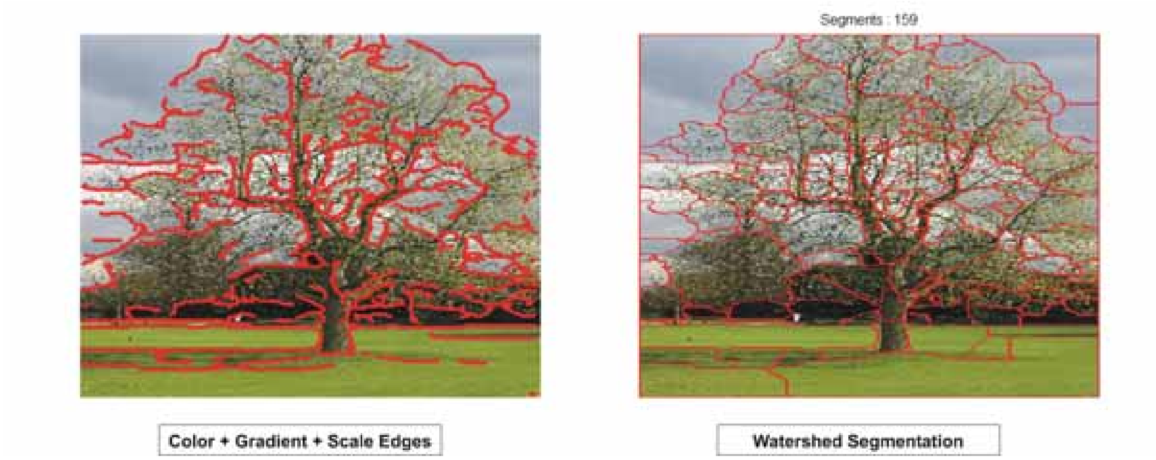


Figure 2.8 Comparison of different distance transform used for Watershed Segmentation

We use the implementation of watershed segmentation algorithm approach represented by a seminal paper [28]. The algorithm is applied on the negative of the distance transform on the detected edges and is described as resulting from a flooding simulation. Given the edge map, the map of distances from edges is modeled as a plastic surface where holes are created on valleys. Each minimum in the distance map represents one catchment basin, and the strategy is to start at the altitude minima. Imagine that there is a hole in each local minimum, and that the topographic surface is immersed in water. As a result, the water starts filling all catchment basins, minima of which are under the water level. If two catchment basins would merge as a result of further immersion, a dam is built all the way to the highest surface altitude and the dam represents the watershed line. As can be seen from Figure 2.8, different distance functions gives different initial segmentations starting from the same edges. The chessboard function gives the best result and the minimum number of initial segments, which we think is appropriate for our approach. The disadvantage of using other distance function is that it results in too much of over segmentation and merging criteria becomes more complex afterwards. Figure 2.9 shows the initial segmentation obtained for the tree image.



Color + Gradient + Scale Edges

Watershed Segmentation

Figure 2.9 Watershed Segmentation

2.8 Initial Merging

After an image is segmented, its adjacent regions can be merged to form a less fragmented segmentation based on their similarity. Similarity metrics can be defined as a function of intensity, color, size, texture, variance, and shared border ratio, among others. In addition, for any feature, different techniques can be derived to measure similarity. For instance, the difference in intensity between adjacent objects can be measured by two methods. In one method, the average of all the pixel intensities in one object can be compared to the average intensity of all the pixels in an adjacent object. Alternatively, local contrast can be computed by comparing the average intensity made up of just the pixels that are adjacent to the border between two objects. One or the other or the maximum of the two can be used. For another alternative, the size of the objects being compared could be used to select which contrast method is employed. Global contrast can be computed when small objects are being considered, and local contrast can be used for large objects. A second method would be to select the contrast computation method based on the variance of the object intensities. Where large intensity variance exists, local contrast might be more appropriate, whereas global contrast could be used for objects that have low intensity variance.

We use local contrast at the border of the segments to merge segments which are smaller than a certain threshold (< 1% of the total pixels of the image). It should be noted that some mistakes in merging can be made at this step, but it do not affect the global result. This allows comparing only bigger segments for which histogram distributions are more reliable in the next merging step. As shown in Figure 2.10 around nineteen segments are reduced after this step.

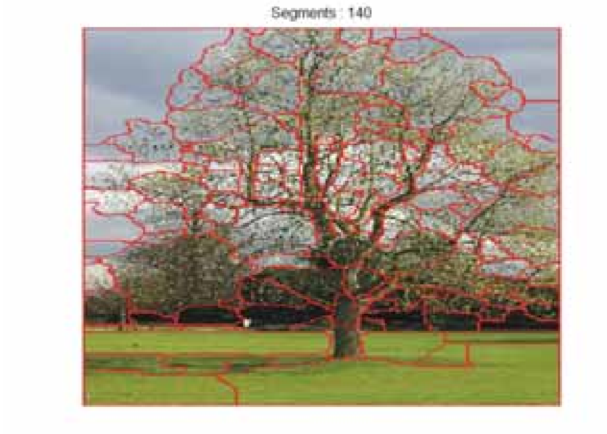


Figure 2.10 Initial Merging to eliminate small segments

2.9 Elimination of High Scale Edges

By carefully observing the scale maps of many images, we observed that the salient boundaries have a very fine scale. This fact we use to find false segment boundaries, which are generally formed due to noisy edge pixels during the watershed segmentation. The idea is to find the mean scale for the boundary, and merging those segments which have a high scale boundary in common. As seen from Figure 2.11B the boundaries with mean > 6 are eliminated.

Figure 2.11 Eliminating perceptually insignificant edges

2.10 Link Merging

There exist many merging algorithms. A general approach to region merging segmentation is given by the following algorithm [29], and generally needs a region adjacency graph (RAG) to be created first.

1. Define some starting method to segment the image into many small regions.
2. Define a criterion for merging two adjacent regions.
3. Merge all adjacent regions satisfying the merging criterion. If no two regions can be merged, stop.

The result of region merging usually depends on the order in which regions are merged, meaning that segmentation results will probably differ if segmentation begins, for instance, in the upper left or lower right corner. This is because the merging order can cause two similar adjacent regions R1 and R2 not to be merged, since an earlier merge used R1 and its new characteristics no longer allow it to be merged with region R2. If the merging process used a different order, this merge may have been realized.

Our approach is motivated from [8;17]. Instead of using a graph cut approach we implemented a simple "Link method" as shown in Figure 2.12. Our algorithm stresses on aggregation. Instead of merging two regions as above, we construct a node graph in the first step. We assume every segment as one node. Segments which are adjacent to each node and which are similar, (the similarity criteria are mentioned later) are connected to that node (Comparison step). A node can be empty if there are no similar segments or may contain more than two segments. In the aggregation step, the first node is taken and the segments connected to this are added to this node, once a node is added to another node its connection is made empty. Subsequent nodes are added till all nodes connected to first node become empty in the node graph. Now the algorithm moves to next non-empty node and above procedure is repeated. This step stops till it reaches the last empty node. In the last step the segments which are present in non-empty nodes are merged. This stage is repeated another time, to make a multi-scale aggregation.

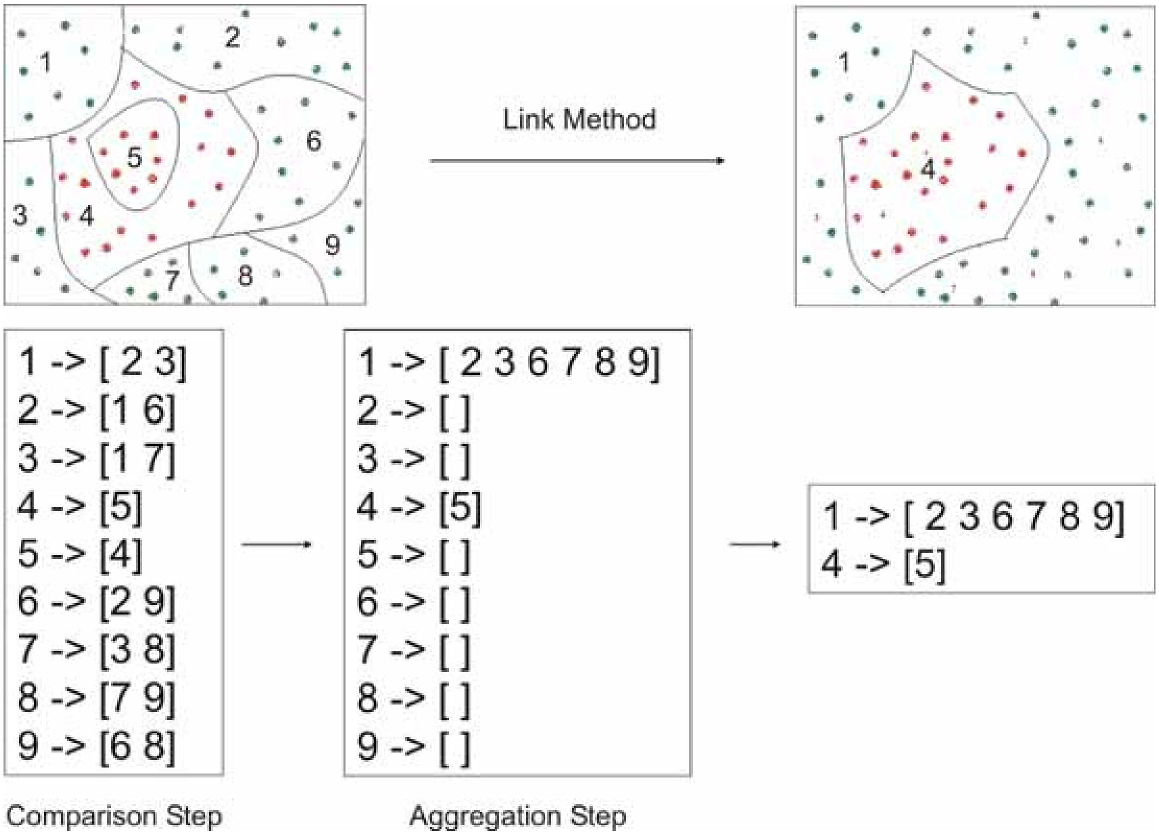


Figure 2.12 Link method for merging

The main advantage of this new method is that it does not depend on the starting point of the node. So the final aggregation is same even if the algorithm is run again. Another advantage is that it also solves the problem of illumination to some extent. Consider for example Figure 2.13 the segments of the tree which are in front of sky are bright, while which are in front of other trees are dark, but intermediate segments which are quite similar and linked to both allows the merging of the lighter and darker segments of the tree, which would have been difficult if the merging was done in a pair-wise traditional way.

To compare the segments we employ the 1976 CIE L*a*b* color space. We model brightness and color distributions with the joint L*a*b* histogram by binning each channel with 64 bins. This is motivated by the fact that a* and b* correspond to the green-red and yellow-blue color opponent channels in the visual cortex, as well as the perceptual orthogonality of the two channels (see [4]). For the texture, we consider the scale distributions from the scale map and rotational scale distribution for the rotational scale map with 4 and 6 bins respectively. The histograms are compared with χ^2 histogram difference operator.

2.11 Final Merging

The final merging uses a new illumination invariant similarity measure between histograms as described in [14]. Input image is converted into the c1c2c3 illumination invariant color space. Color histograms h_i[c1,c2,c3] are computed with 64 bins. Cross-correlation between histograms of segments for each color channels is done separately and the maximum values of cross-correlation in some range (t1, t2).

s = (s1, s2, s3) such that s1 ≤ s2 ≤ s3 are sorted maximum values of the cross-correlations. The squared distance of two regions i & j is then evaluated as $d(i, j) = (s1 - s2)^2 + (s3 - s2)^2$. The segmentation after final merging is shown in Figure 2.13

Figure 2.13 Segmentation after Link merging and the final merging steps

2.12 Results

The results of our segmentation algorithm are shown in Figure 2.14. We tested our algorithm on the Berkeley dataset [3] which contain images from natural scenes with a variety of texture. Figure 2.15 shows the versatility of our algorithm in segmenting gray scale images. Here only brightness and the two scale maps are used.

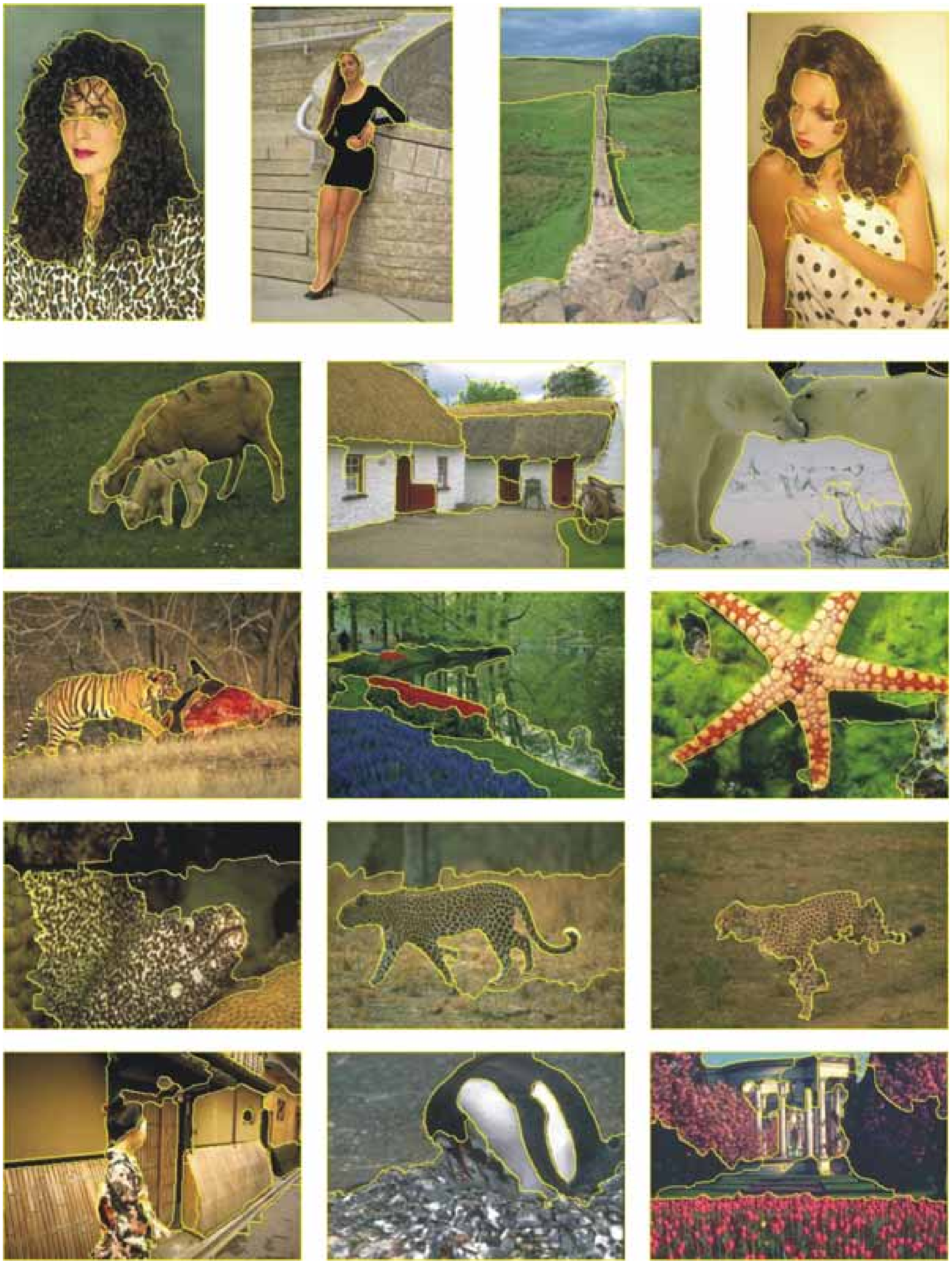


Figure 2.14 Some segmentation results on the images in Berkeley dataset

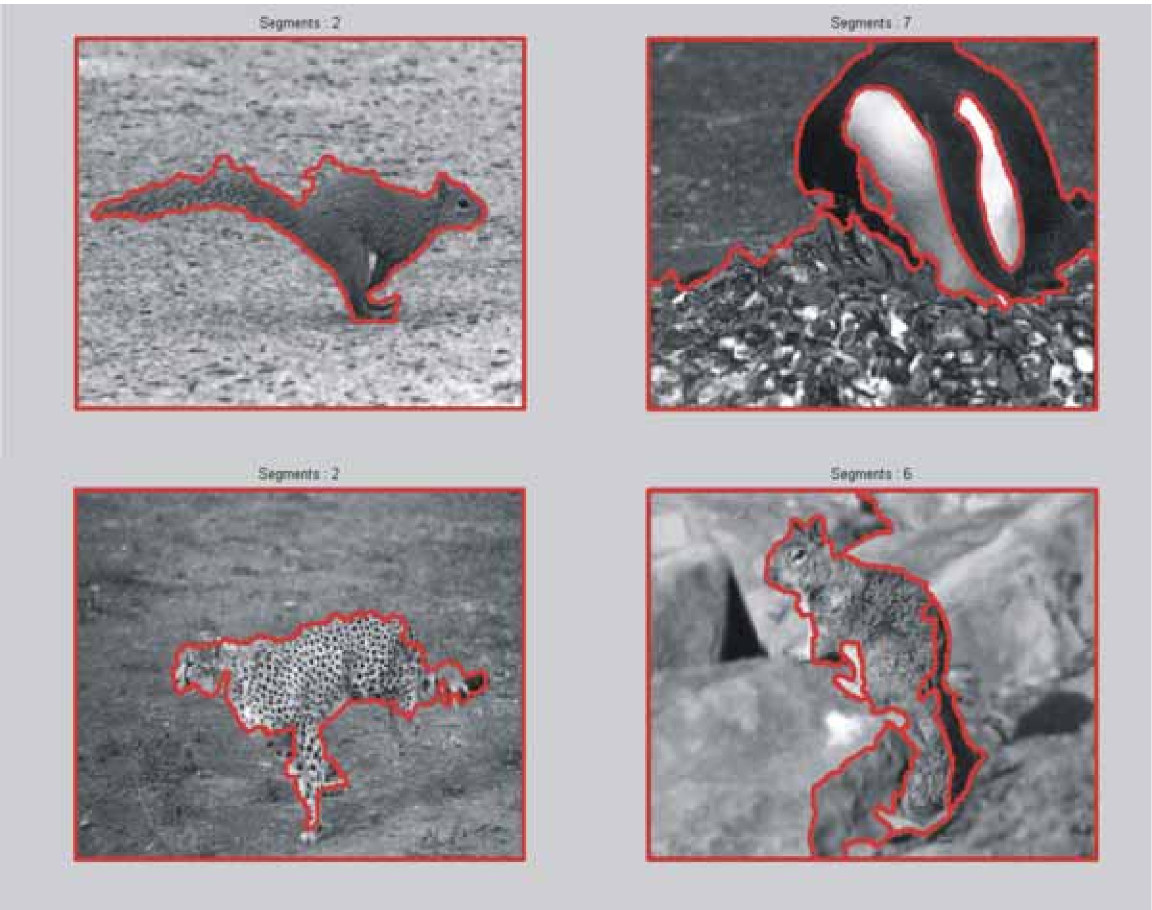


Figure 2.15 Some segmentation results on the gray scale images

2.13 Discussion

Take the example of the last woman in row 1, her dress/towel contains some small dots, our algorithm merges some dots into the hand, this are the mistakes at the small segment merging as described before, as there is no knowledge whether this dots belong to hand or towel. However it is to be noted that this mistakes didn't affect the global result on that image.

As said earlier the scale information is useful only for the prediction of presence or absence of texture, it is not robust in separating two textures with similar scale distribution which is the case of camouflage. We believe that, to segment camouflage properties such as distance from the centre of focus, should be used, along with scale information. A typical example is shown in Figure 2.15 where the chicks of penguin are merged into the stony textured ground.

The algorithm's major time is spent in edge detection. Watershed algorithm takes less than half a second. Merging is done in 5-10 seconds. The mean total time calculated over the 200 images in Berkeley database is 33 seconds.

“If I'd asked people what they wanted, they would have asked for a faster horse.”

……Henry Ford

3 Image Segmentation Benchmark

It is considerably easier to quantify the performance of computer vision algorithms at recognition than at segmentation. Recognition is classification, and one can empirically estimate the probability of misclassification by simply counting classification errors on a test set. It is well accepted that one cannot evaluate a recognition algorithm by showing a few images of correct classification. In contrast, image segmentation performance evaluation remains subjective [3]. Typically, researchers will show their results on a few images and point out why the results 'look good'. We never know from such studies whether the results are best examples or typical examples, whether the technique will work only on images that have no texture, and so on.

3.1 Berkeley Database

To benchmark the results of the algorithms, we made use of the Berkeley segmentation benchmark described in [3], In this study a database containing 'groundtruth' segmentations produced by humans for images of a wide variety of natural scenes is presented. It was noted that the segmentations produced by different humans for a given image are not identical, but, were consistent. One can think of a human's perceptual organization as imposing a hierarchical tree structure on the image. Even if two observers have exactly the same perceptual organization of an image, they may choose to segment at varying levels of granularity. This implies to define segmentation consistency measures that do not penalize such differences. Two measures of the difference between two segmentations S1 and S2 are introduced in this paper, the Global and Local Consistency Errors (GCE and LCE). As the GCE is a more demanding measure, we make use of only this measure. They demonstrated empirically that human segmentations for the wide variety of images in the database are quite consistent according to these criteria, suggesting a reliable standard to evaluate different computer algorithms for image segmentation.

It should be also noted that in a later study a methodology for evaluating the performance of boundary detection techniques with this database was developed in [27]. It is based in the comparison of machine detected boundaries with respect to human-marked boundaries using the Precision-Recall framework, a standard information retrieval technique. Precisely, two quality measures are considered, Precision (P), defined as the fraction of detections that are true boundaries and Recall (R), given by the fraction of true boundaries that are detected. Thus, Precision quantifies the amount of noise in the output of the detector, while Recall quantifies the amount of ground-truth detected. Measuring these descriptors over a set of images for different thresholds of the detector provides a parametric Precision-Recall curve. The two quantities are then combined in a single quality measure, the F-measure, defined as their harmonic mean: Finally, the maximal F-measure on the curve is used as a summary statistic for the quality of the detector on the set of images. As our segmentation output does not give boundary probabilities we do not evaluate our segmentation using this measure.

3.2 Global Consistency Error & Local Consistency Error

Segmentation is simply a division of the pixels of an image into sets. A segmentation error measure takes two segmentations S_1 and S_2 as input, and produces a real valued output in the range [0::1] where zero signifies no error. For a given pixel p_i consider the segments in S_1 and S_2 that contain that pixel. The segments are sets of pixels. If one segment is a proper subset of the other, then the pixel lies in an area of refinement, and the local error should be zero. If there is no subset relationship, then the two regions overlap in an inconsistent manner. In this case, the local error should be non-zero. Let \ denote set difference, and $|x|$ the cardinality of set x. If $R(S, p_i)$ is the set of pixels corresponding to the region in segmentation S that contains pixel p_i, the local refinement error is defined as:

$$E(S_1, S_2, p_i) = \frac{|R(S_1, p_i) \setminus R(S_2, p_i)|}{|R(S_1, p_i)|} \qquad \textbf{eq. 2}$$

Note that this local error measure is not symmetric. It encodes a measure of refinement in one direction only: $E(S_1, S_2, p_i)$ is zero precisely when S_1 is a refinement of S_2 at pixel p_i, but not vice versa. Given this local refinement error in each direction at each pixel, there are two natural ways to combine the values into an error measure for the entire image. Global Consistency Error (GCE) forces all local refinements to be in the same direction. Local Consistency Error (LCE) allows refinement in different directions in different parts of the image. Let n be the number of pixels:

$$GCE(S_1, S_2) = \frac{1}{n} \min\left\{ \sum_i E(S_1, S_2, p_i), \sum_i E(S_2, S_1, p_i) \right\} \qquad \textbf{eq. 3}$$

$$LCE(S_1, S_2) = \frac{1}{n} \sum_i \min\left\{ E(S_1, S_2, p_i), E(S_2, S_1, p_i) \right\} \qquad \textbf{eq. 4}$$

As $LCE \leq GCE$ for any two segmentations, it is clear that GCE is a tougher measure than LCE. Note that since both measures are tolerant of refinement, they are meaningful only when comparing two segmentations with an approximately equal number of segments. This is because there are two trivial segmentations that achieve zero error: One pixel per segment, and one segment for the entire image. The former is a refinement of any segmentation, and any segmentation is a refinement of the latter.

3.3 Comparison with Other Methods

The current public version of the Berkeley database is divided in two independent sets of images. A training set of 200 images and a test set of 100 images. We used the 200 color images in the test group of the Berkeley Segmentation Dataset as well as the corresponding human segmentations. For each of the images, at least 5 segmentations produced by different people are available. As GCE is a tougher measure we use it for benchmarking the results.

Method	# of regions	Global Consistency Error
Human	17	0.08
Our Method	**11**	**0.173**
Seed	4	0.209
Max-Sum Solver	8	0.178
Fow	13	0.214
Ncuts	5	0.336

Table 1. Comparison with other methods

For each image, the GCE of the segmentation produced by the tested algorithm with respect to each of the available human segmentations for that image was calculated. The mean of these values gives the mean GCE per image, which was plotted in a histogram, see Fig. 8. The global GCE was calculated as the mean of these 200 mean GCE values. We compared human segmentations to each other and then with the normalized cuts algorithm (Ncuts) [10], (Fow) [11], seed algorithm (seed) [14] and max sum solver [15]. Comparison of human vs. human produces a very low GCE value which indicates the consistency of the human segmentations. The benchmarked results other than ours are taken from [14;15] See Table 1 for the results.

“If you tell the truth, you don't have to remember anything.”

…………Mark Twain

4 Shape Extraction & Classification

Shape recognition has been used in the community starting from the silhouettes. Mammal classification is a challenging topic and has started receiving significant attention lately. The main aim of this chapter is to test if our segmentations are indeed robust to extract shapes and perform recognition using these shapes.

4.1 Mammal Dataset

We collected 100 images of 6 classes of mammals (Cat, Dog, Cow, Camel, Horse and Sheep). Horse images are taken from Weizmann Horse dataset [30]. Some of the images of cow are taken from MSRC database [31], the rest of the images are obtained by performing multi-lingual Google image search, by this we mean that to search for Dog we typed Cane - in Italian, Hund – in German, Kutta- in Hindi and so on. There exists another work which has collected a labeled mammal dataset [32]. However in this dataset there are too many classes and there exist no perfect segmentations for them, only a rectangle is drawn around the animal. Also we don't want to make the task more difficult. We collected a limited class because first we wanted to check whether bottom-up segmentation method can extract the shape of the object reliably and to see if shape is indeed a key feature for classification of inter-class object category. Figure 4.1 shows some examples of each class. We manually selected the contour of the mammals to obtain perfect segmentation; this will allow us to see the best classification result that can be obtained using shape. Figure 4.2 shows one example of each class and marked silhouettes. We further marked manually the best segments from our final segmentations on all the images in this dataset. We wanted to test what will be the best classification result assuming that we had a perfect shape extraction algorithm. Figure 4.3 shows the shape extracted manually after applying our segmentation algorithm.

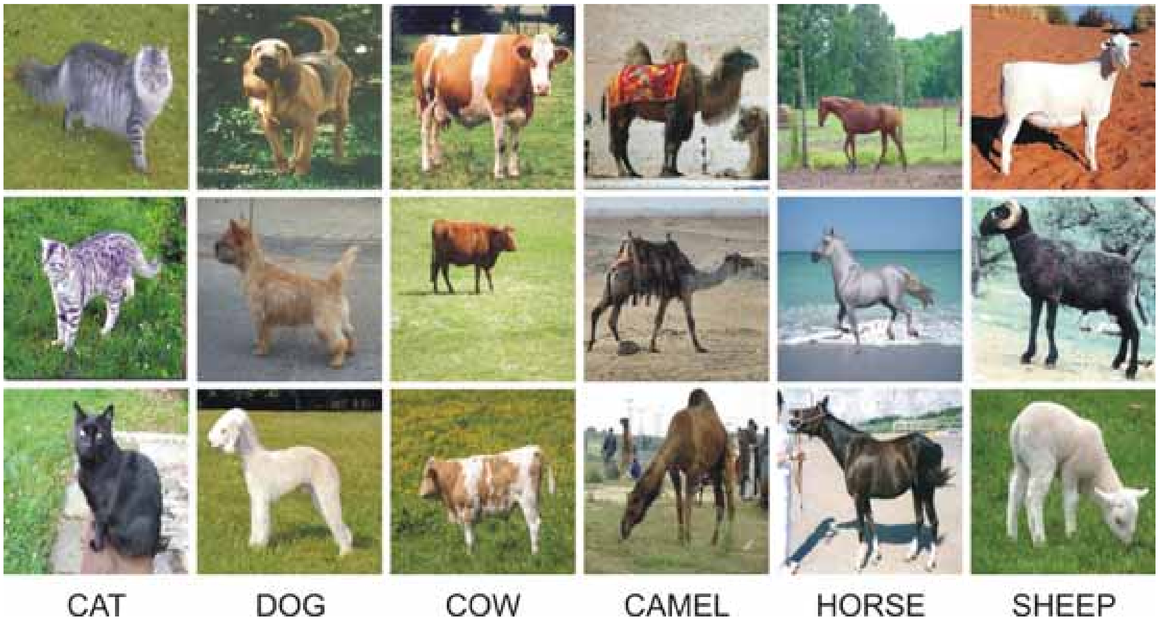


Figure 4.1 Sample Images from the Mammal Database

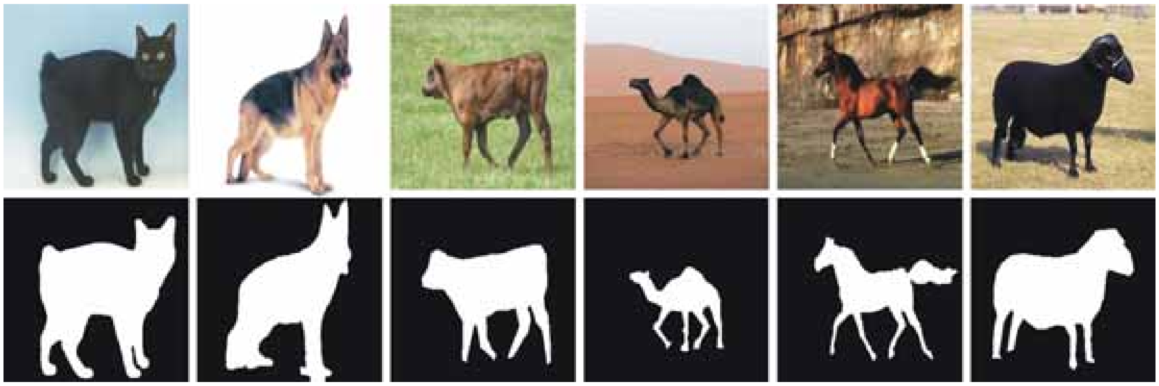


Figure 4.2 Manual Segmentation done on the Mammal Database

Figure 4.3 Shape from automatic segmentation done on the Mammal Database

4.2 Shape Extraction Method

First we segment the image using our algorithm. The final segmentation can vary from 2-20 segments. It is not always possible to obtain a perfect figure-ground separation due to several factors, such as shading, illumination, and presence of more structures around like sky, water etc. We assume that distance from the centre of focus plays a key role in segmentation. We use distance transform applied from the centre of the image. First order moments are calculated for each segment, i.e. the sum of the distance from the centre for each pixel in that segment normalized with the area of the segment. The segments with higher distance from the centre are obviously background segments, which can be seen from Figure 4.4.

However a part of the object can be isolated and may have a higher distance, like the tail for example. So we find another map where we find the saliency of the segments using the following algorithm

1. For each segment "A(i)" find the segments B(j) surrounding it and also the length of shared boundary.
2. For each segment in set B find the saliency using the formula below

$$\text{Saliency(i)} = \text{total\_boundary\_A(i)\_and\_B} * \sum_{j} \text{length\_of\_shared\_boundary\_A(i)\_and\_B(j)} * \left(\text{Dist\_seg\_B}(j) - \text{Dist\_seg\_A}(i)\right)$$

, where "Dist_seg" is the mean distance for that segment found from the previous map.

Thus for segments closer to the centre the saliency will be higher, while for the boundary segments it will be higher. Using the saliency map and moment map we find a combination of shapes as shown in Figure 4.5 top row. The first shape uses the combination of both saliency map and moment map, the second shape is obtained by thresholding the moment map and the third shape by thresholding the saliency map.

Figure 4.4 Shape extraction

4.3 Classification Scheme

The classification scheme is shown in the Figure 4.5. For each image in the test set, we have a combination of shapes extracted from the above section. We then compare each segment with the training shapes by computing pair-wise similarity and find the 5 most similar shapes for each extracted shape, thus in this case we have 15 matches. We then find the class which is retrieved the maximum number of times and classify the given image with it. The details for the comparing shapes are shown in Figure 4.6. We use the method from [33]. The algorithm analyzes the contour of pairs of shapes. Their contours are recovered and represented by a pair of N points obtained by linear interpolation. Given two points pi and qj from the two shapes the cost of their matching is evaluated by using the shape context and by using dynamic programming the best matching between the point sets is obtained. Dynamic programming not only recovers the best matching, but also identifies occlusions, i.e. points in the two shapes which cannot be properly matched. Given the correspondence between the two point sets, the two contours are aligned using Procrustes analysis. After alignment, each contour is transformed into a string of symbols and a modified version of edit distance is used to compute the similarity between strings of symbols. Finally, is obtained by a simple nearest-neighbor procedure.

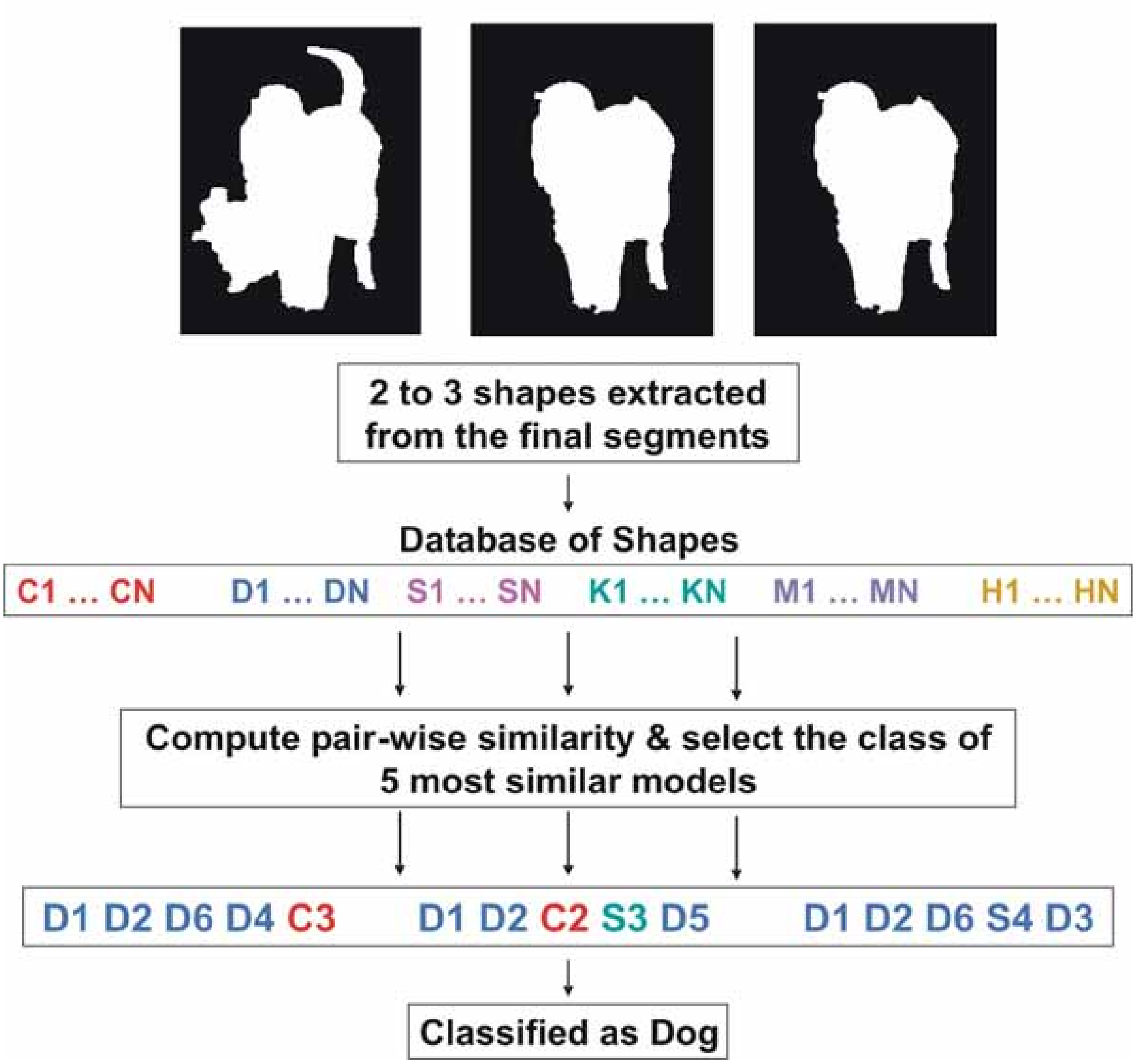


Figure 4.5 Shape classification scheme

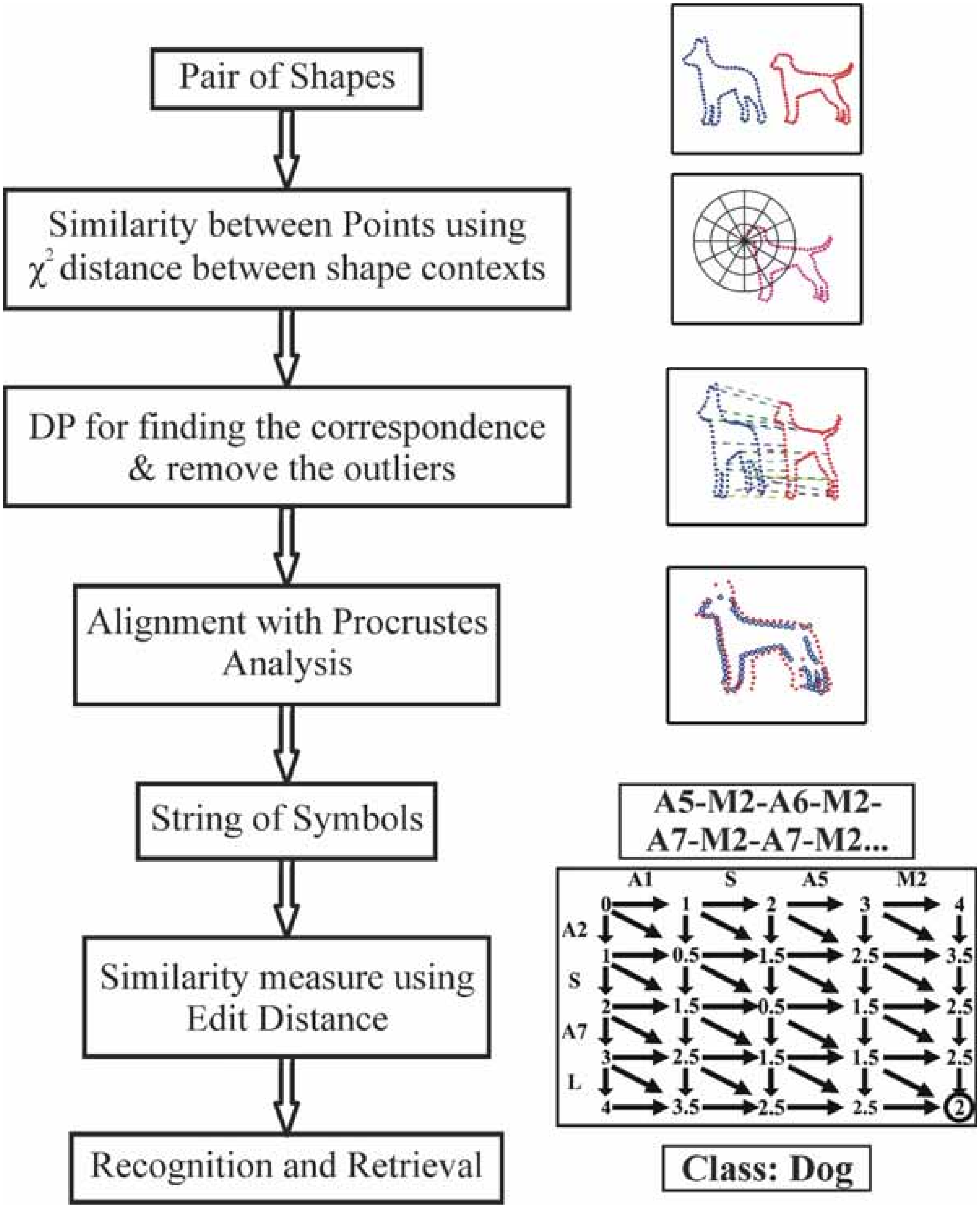


Figure 4.6 Shape classification algorithm

4.4 Comparison

We did the classification on our mammal dataset using the procedure described in the previous sections. Figure 4.7 shows the comparison of recognition rate done with different methods. The graph shows the recognition rate using different training set size. For the "Perfect Segmentation" method we used the manually extracted shape as described in section 4.1. For "Manual 1" the test set consists of shapes that are extracted using the best segments selected by hand from our final segmentation and the training set used the perfect shapes. For "Manual 2" both the training and test set consists of shapes that are extracted using the best segments selected by hand from our final segmentation. The "Automatic" the test set uses the shape extraction method in section 4.2 and the training set uses the perfect shapes. We compared the results with the famous "Bag of features" [34;35] approach using the online code (http://vision.ucla.edu/~vedaldi/code/bag/bag.html) which uses SIFT features. Also shown is the classification by chance, obtained after a random class is assigned to the image. The classification is repeated for 100 times to randomize the test and training test. The classification rate is the mean over this 100 runs. The classifier used is K-NN, with K=5, except in case where training set was 5, we use K=3;

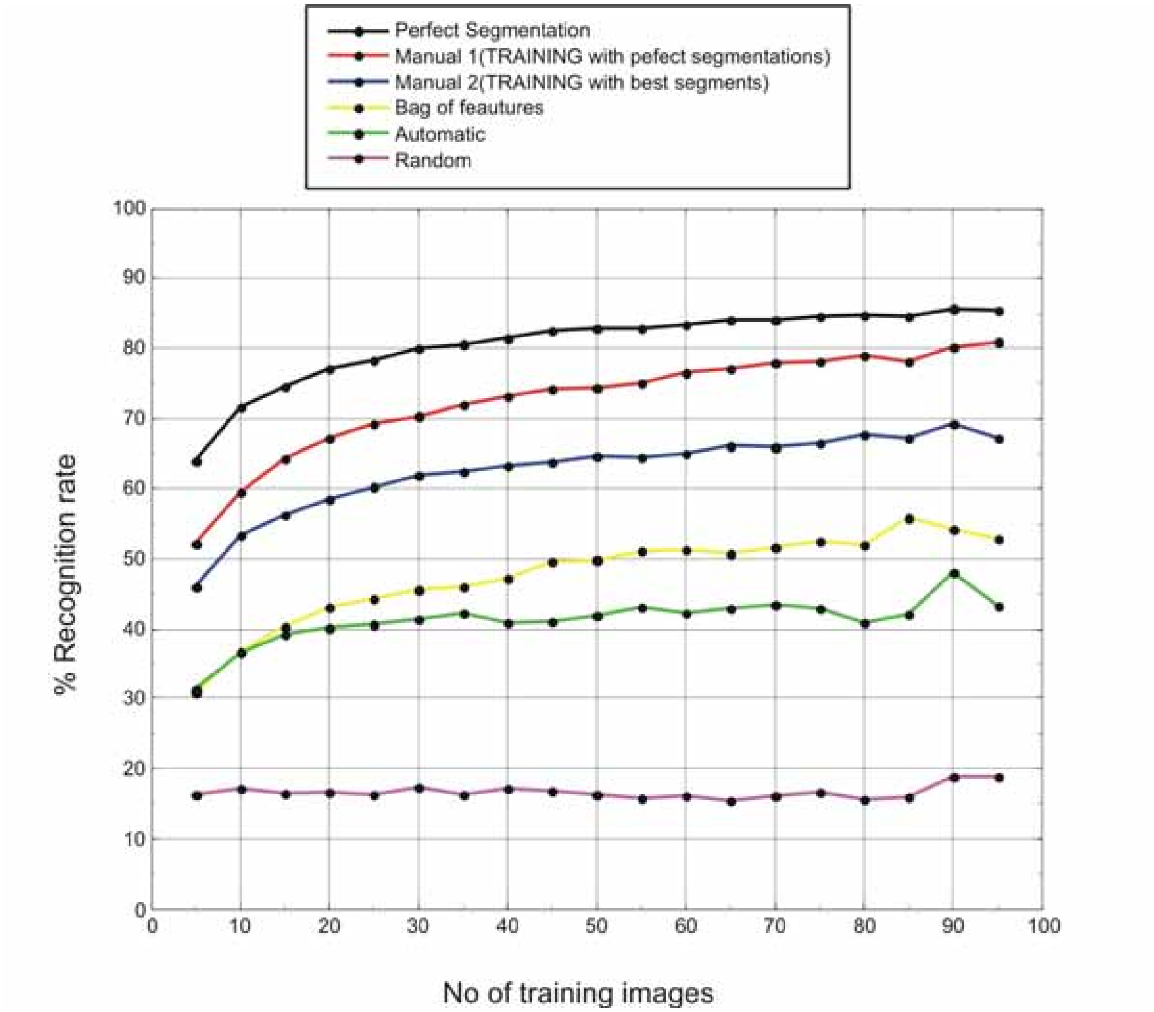


Figure 4.7 Results on Mammal Database

Figure 4.8 shows the individual classification results for each class. Also shown is the confusion matrix for the classes. Figure 4.9 shows the improvement in classification if texture is combined along with shape. For texture feature we used only three circular filters as described in the next chapter.

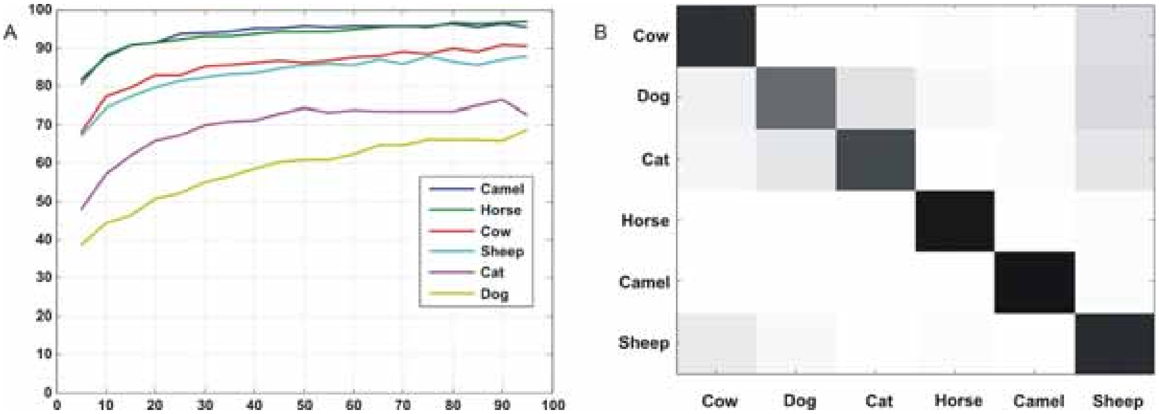


Figure 4.8 Individual recognition rate for each class and Confusion Matrix

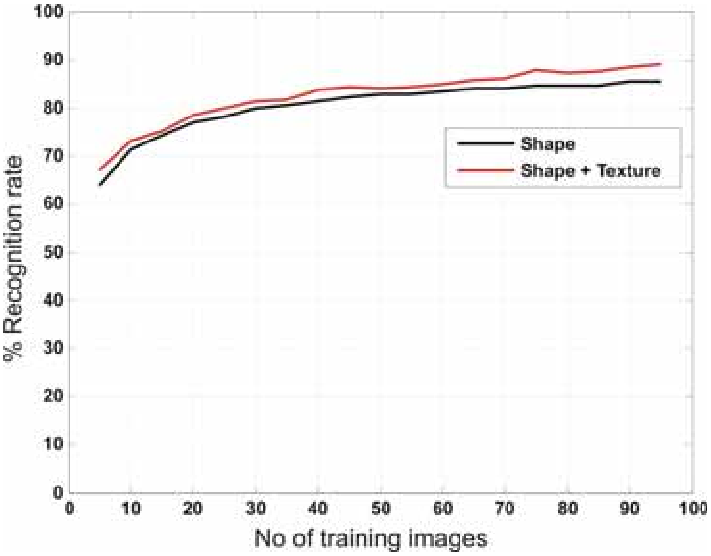


Figure 4.9 Results when texture information is combined

4.5 Discussion

The most difficult category is the Dog, which is followed by Cat, describing their variability of shapes. Though the "bag of feature approach" is a bit better than our automatic segmentation, it should be noted that the "bag of feature approach" does not provide segmentation of the object, as our method does. There is also a big chance for improvement if the shape extraction algorithm is modified taking into some global characteristic of shape. Texture does not make a significant improvement to the result. We didn't test the results with color, but we believe that it should not add more information; on the other hand it may worsen the results. It might improve the result on camel and sheep though which have less color variability.

“The important thing in science is not so much to obtain new facts as to discover new ways of thinking about them.”

…….Sir William Bragg

5 Texture Classification

Texture is a very broad area and covers a huge range of concepts. However within this thesis we will exclusively deal with texture captured within images. Textures both synthetic and artificial are abundant within the real world and so it is essential that computer vision systems can successfully deal with textural information present in images. Texture in brief can be divided into two types uniform and non-uniform. Research in this area is important within the vision community as any application which deals with image information will inevitably contain a significant amount of textural information. Texture Research is divided into four areas [36]: (1) synthesis; (2) classification; (3) segmentation; and (4) shape from texture. In this thesis we focus only on classification.

5.1 State of Art

Previous approaches mainly used statistical representations of the responses of bank of filters [1;37-41]. These filter response distributions were learned from training images and represented by clusters [1;38;41;42] or histograms [39;43]. These distributions are then used for classification, segmentation or synthesis.

Distribution-based methods have been introduced for classifying 3D textures under varying poses and illumination changes. The basic idea is to compute a texton histogram based on a universal representative texton dictionary. Leung and Malik [1] constructed a 3D texton representation for classifying a stack of registered images of a test material with known imaging parameters. The special requirement of calibrated cameras limits the usage of this method in most practical situations. This limitation was removed by the work of Cula and Dana [38], who used single-image histograms of 2D textons. Varma and Zisserman [41;44] have further improved 2D texton-based representations, achieving very high levels of accuracy on the Columbia-Utrecht reactance and texture (CUReT) database [45]. The descriptors used in their work are filter bank outputs [41] and raw pixel values [44]. Hayman et al. [46] extend this method by using support vector machine classifiers with a kernel based on $\chi 2$ histogram distance. Even though these methods have been successful in the complex task of classifying images of materials despite significant appearance changes, a major shortcoming of these methods is that the underlying representation is not geometrically invariant. No adaptation is performed to compensate for changes in scale or surface orientation with respect to the camera. Our approach is close to [41] as we also use filter bank and texton representation, the main difference is the generation of texton dictionary which in our case is unique and independent of dataset or training set used. We have used only rotational invariant filters.

Lazebnik et al. [47] proposed an intrinsically invariant representation based on distributions of appearance descriptors computed at a sparse set of affine-invariant keypoints. The Harris-Laplace detector responds to corner-like regions, while the Laplacian detector extracts blob-like regions. Zhang et al. [48] further improved this work by combing different channels and descriptors. A combination of SIFT [49], SPIN and RIFT [47] descriptors have been used. Support Vector Machine (SVM) classifiers [50] have shown their promise for visual classification tasks, and the development of specialized kernels suitable for use with local features have significantly improved the classification task [51]. Previous comparisons and evaluations of methods combining kernels and local features were limited to one or two datasets. A comprehensive study for classification accuracy of five different methods for 4 texture databases was recently performed by [48].

5.2 Texture Datasets

For our experimental evaluation, we use the same four texture datasets used by [48].The texture datasets are UIUCTex [47], KTH-TIPS [46], Brodatz [52], and CUReT [45]. The Brodatz texture album is a famous benchmark dataset. It contains 112 different texture classes where each class is represented by one image divided into nine sub-images [48]. This dataset does not model viewpoint, scale, or illumination changes. Sample images are shown in Figure 5.1

For the CUReT texture database, we use the same subset of images as [13, 20]. This subset contains 61 texture classes with 92 images for each class. These images are captured under different illuminations with seven different viewing directions. The changes of viewpoint and the illumination direction, can affect the texture appearance. Sample images are shown in Figure 5.2

The UIUCTex dataset contains 25 texture classes with 40 images for each class, viewed under significant scale and viewpoint changes. The dataset includes non rigid deformations, illumination changes and viewpoint-dependent appearance variations. Figure 5.3 presents two sample images per class, each showing a textured surface viewed under different poses.

The KTH-TIPS dataset contains 10 texture classes. Images are captured at nine scales spanning two octaves (relative scale changes from 0.5 to 2), viewed under three different illumination directions and three different poses, giving a total of 9 images per scale, and 81 images per material. From this figure, we can see that scaling and illumination changes can increase the intra-class variability and reduce the inter-class separability, which can increase the difficulties of the classification task. Sample images with scale and illumination changes are shown in Figure 5.4

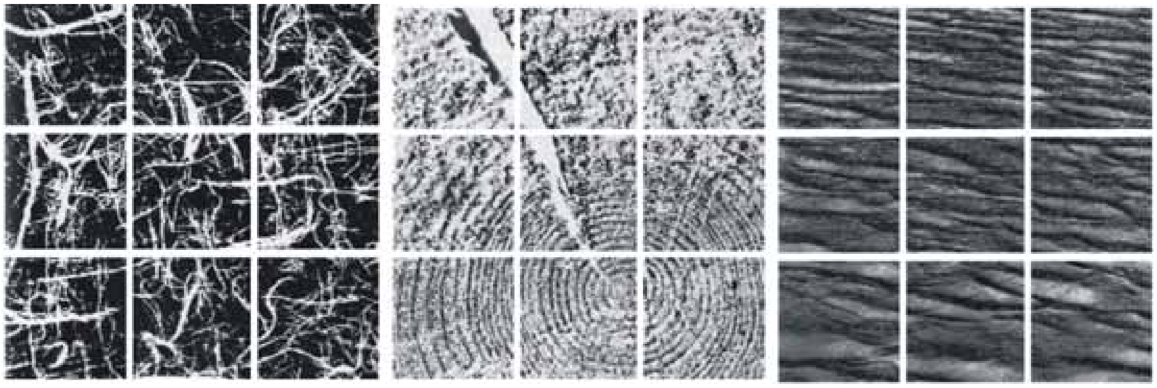


Figure 5.1 Brodatz Dataset

Figure 5.2 CUReT Dataset

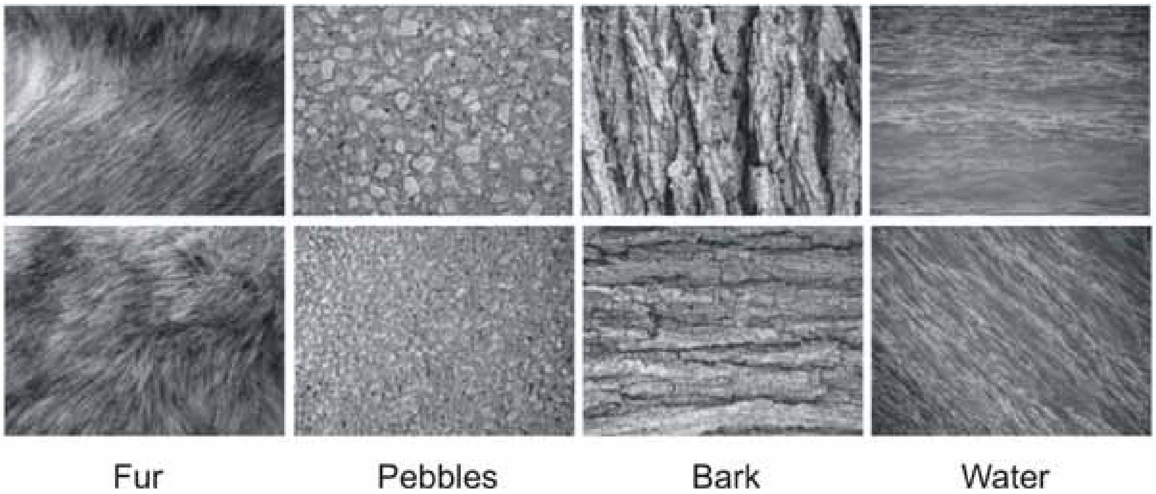


Figure 5.3 UIUCTex Dataset

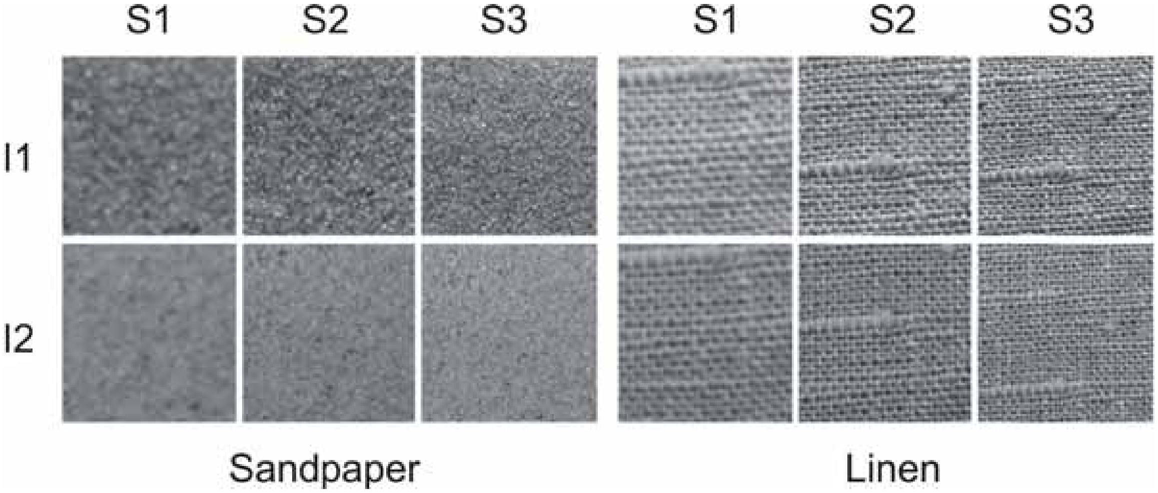


Figure 5.4 KTH-TIPS Dataset

5.3 Overview of the Methodology

The classifier used in this paper does not use color information. Images are normalized, so to achieve partial invariance to changes in light intensity [41]. The algorithm is divided into a learning stage and a classification stage. Training Images are first convolved with a filter bank to generate filter responses. Universal textons are used to label every pixel. The histogram of texton frequencies is used to form models. In the classification stage the same procedure is used to construct the histogram for the test image. The histogram is then classified using SVMs and a $\chi2$ kernel.

5.4 Filters

A recent comparison [41] of filter banks (including those used by [1;38;53], demonstrated that the rotationally invariant, MR8 filter bank [41] yields better results. For this reason rotationally invariant filters were selected in this work.

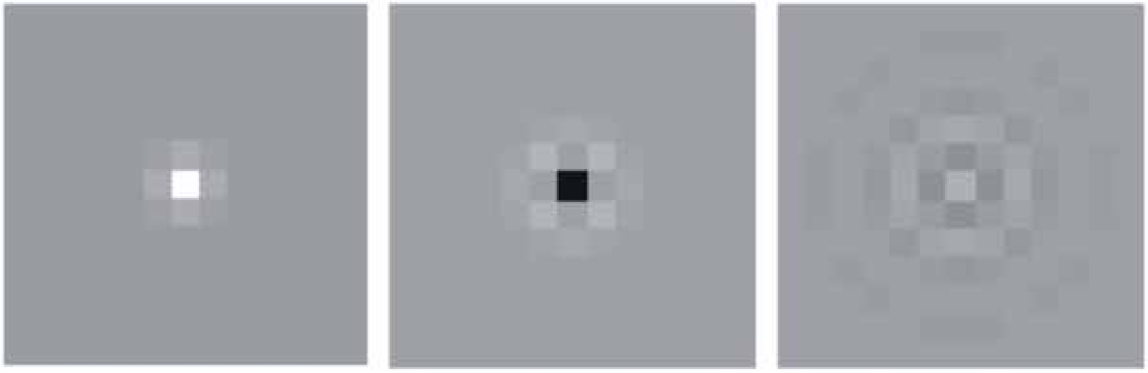


Figure 5.5 The Filter Bank

We use a 13x13 square neighborhood. The first filter is a Gaussian with σ equal 0.5. The second filter is Laplacian of Gaussian with σ equal 0.7. Small values of sigma were chosen in order to capture more details of the texture. The third filter is of the form,

$$F(r,\sigma,\tau) = \cos\left(\frac{\pi\tau r}{\sigma}\right)^{e^{\frac{r^2}{2\sigma^2}}}$$

In our case σ and τ are equal to 2. Filters are shown in Figure 5.5. Image is first convolved with the filters and their response is thresholded so to keep only the positive response. The obtained binary image is referred as the "On" response. In this way we obtain 3 binary images one from each filter. These "On" responses are further convolved with averaging filters of size 3x3 and 6x6, thus obtaining a multi scale representation. Figure 5.6 shows these steps for the first filter of Figure 5.5.

5.5 Preprocessing

Images are converted into a grey scale and intensity is normalized so to have zero mean and unit standard deviation. This normalization gives invariance to global (i.e. across the entire region) affine transformations in illumination. Second, all 3 filters are $L1$ normalized so that responses of each filter lie in the same range: each filter Fi in the filter bank is divided by $\|Fi\|_1$ so that the filter has unit $L1$ norm.

5.6 Universal Textons

The traditional procedure for generating the texton dictionary is to convolve several images from the training set of a particular texture class with a filter bank. Obtained responses are aggregated and clustered into textons using the K-Means algorithm [54]. Textons from different texture classes are combined to form the texton dictionary [41], as shown in Figure 5.7

Our method for generating the universal textons e is different from the traditional procedure. We take advantage that we have only 3 filter responses. The response for the averaging filter on the binary image for a 3x3 neighborhood varies from 0 to 9. Therefore textons can take value from [0 0 0], [0 0 1], [0 0 2] ….[0 0 9]….. to [9 9 9] and the total number of possible textons is, $(\text{Nbhd}^2 + 1)^{\text{no\_of\_filters}}$ where Nbhd is the neighborhood size.

Therefore for a 3x3 neighborhood and 3 filters we have 1000 textons. However most of the textons are similar to each other, hence we want to reduce their number. Therefore we put the first texton in a new set. We add to the new set the texton from the old set which is the farthest from those in the new set. With this procedure we select the first 80 textons and we sort them with respect to the sum of the three dimensions as shown in Figure 5.8. This Universal textons is used for all neighborhoods. We use the same Universal textons for classifying all the four datasets used in the experiments.

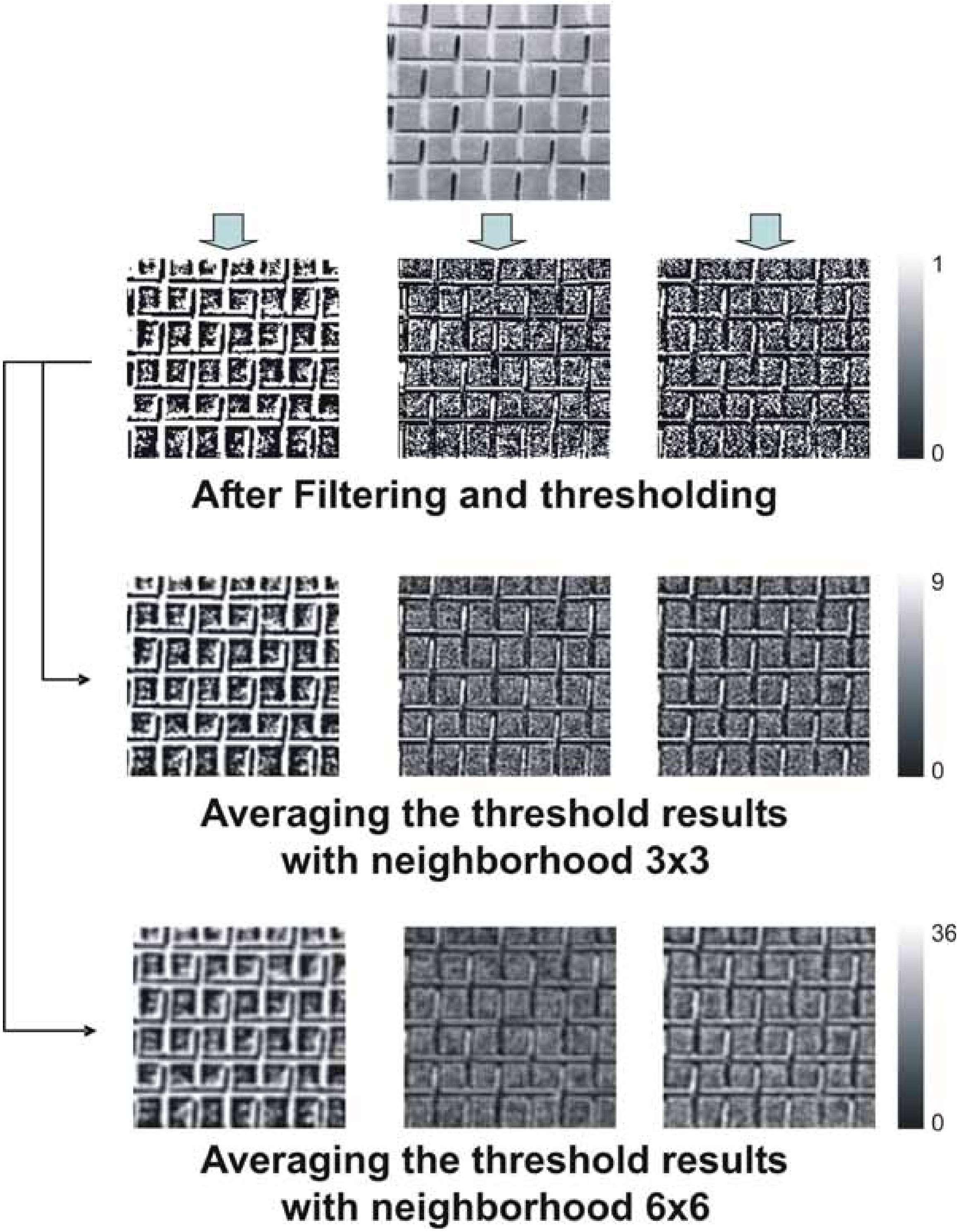


Figure 5.6 Filtering steps

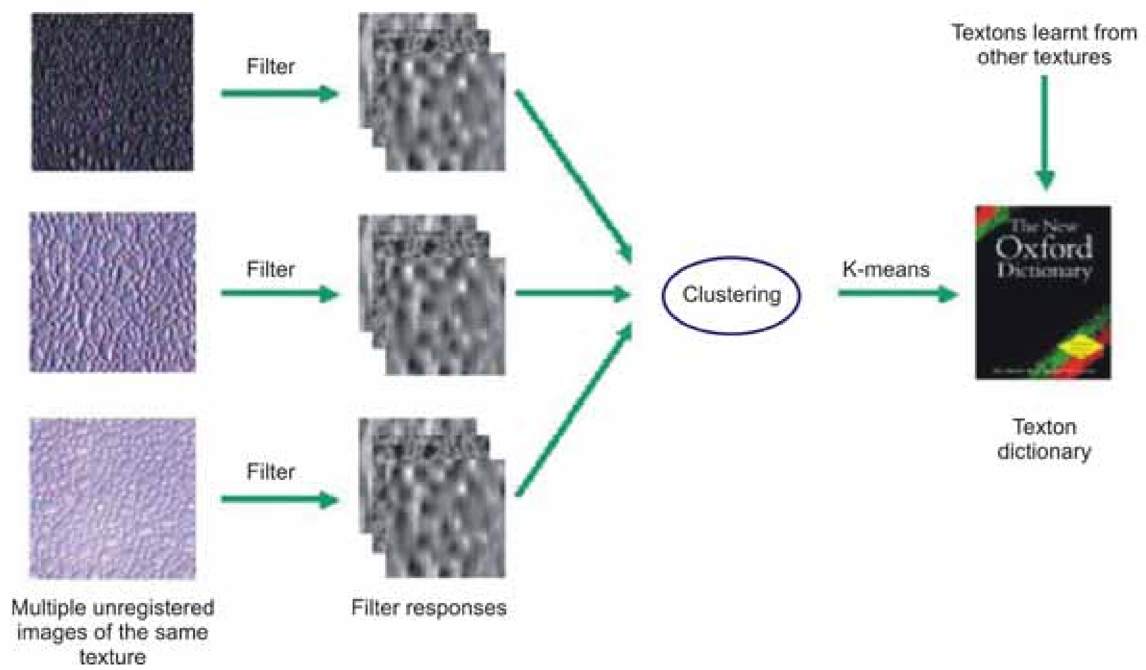


Figure 5.7 Traditional approach for generating Universal Textons

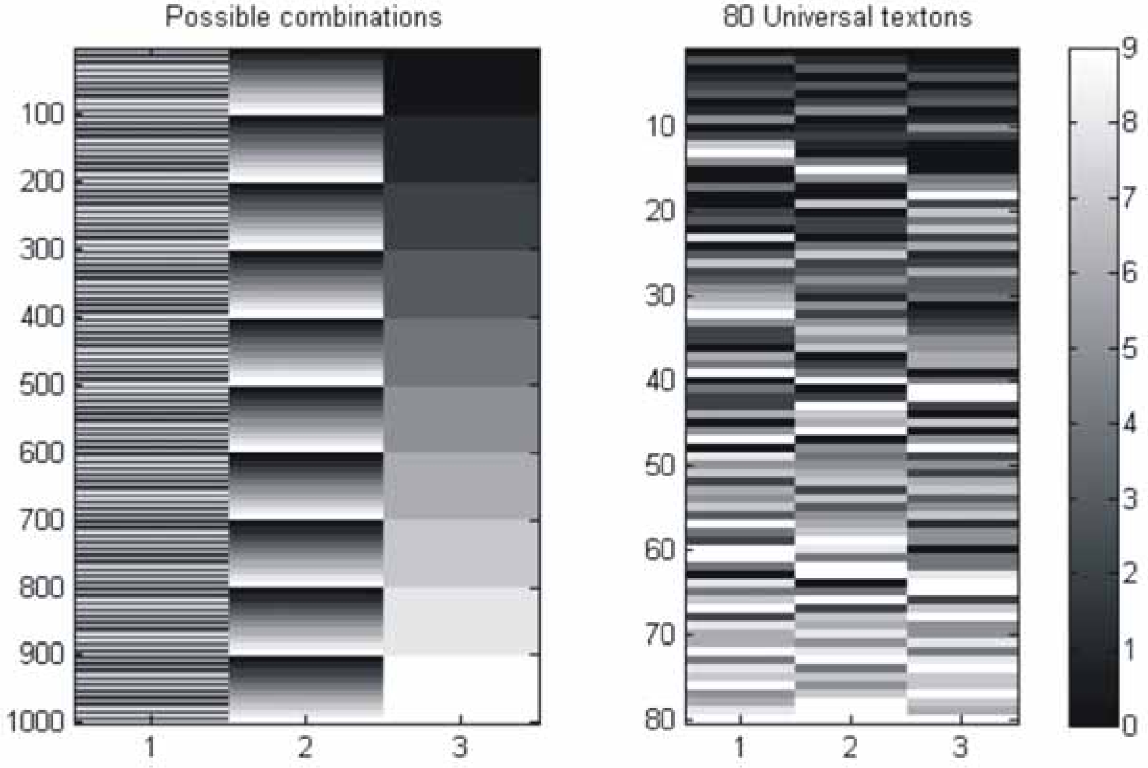


Figure 5.8 Our approach for generating Universal Textons

5.7 Image Representation

For each image we have two representations of textons one for each neighborhood (3x3 and 6x6) as shown in Figure 5.9. These textons capture the fine details of the texture. The model for the image is a 2d histogram combining the information from the two texton maps. Map2 is quantized into 4 overlapping bins. For each pixel represented by this bin, the texton distribution in Map1 is found. The feature vector (Model) is thus represented by 320 dimensions (80 x 4). The Models for two different textures in the KTH_TIPS dataset are shown in Figure 5.9.

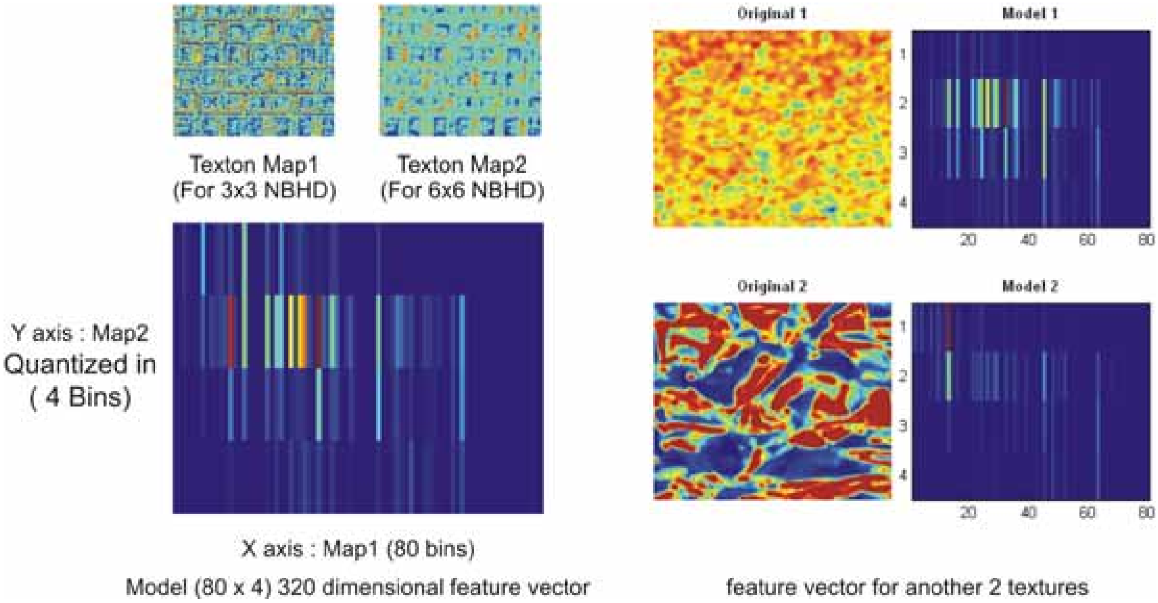


Figure 5.9 Model representation

5.8 Classification Method

To compare two histograms S1 = (u1, . . . , um) and S2 = (w1, . . ., wm), we use the χ2 distance For classification, we use SVMs [50]. Discriminative learning helps to achieve robustness to intra-class variability [48]. We use the matlab implementation of LibSVM [55] available online. The Models from the training set are trained using a χ2 kernel given by $\exp(-\gamma * D(S_i, S_j))$, where $D(S_i, S_j)$ is the χ2 value between Si and Sj histograms. γ is a scaling parameter that can in principle be determined through cross-validation. Setting its value to the mean value of the χ2 distances between all training images gives good results and reduces the computational cost [48].

5.9 Implementation Parameters

We tuned the parameters on a smaller subset of the Brodatz dataset with 20 random texture classes. The training set size per class is 3 and the test set size per class is 6. The first experiment aimed at selecting the number of universal textons. The neighborhood size is kept constant 3x3 and the used feature vector is the frequency distribution of the first texton map. From

Table 2 we see that 80 textons are appropriate and increasing the number of textons does not add more information.

Number of textons	Classification rate
60	90.83
70	90.83
80	**92.5**
90	91.67
100	91.67

Table 2 Number of universal textons.

The second experiment aimed at selecting the size of neighborhood for the first averaging filter. The feature vector is the frequency distribution of the first texton map. The size of the universal textons is chosen from the first experiment using 80 textons3x3 and the used feature vector is the frequency distribution of the first texton map.

Table 3 shows that 3x3 size gives the best classification rate.

Neighborhood size	Classification rate
3	**92.5**
4	91.67
5	91.67
6	91.45

Table 3 Neighborhood size for the first average filter

The third experiment aimed at selecting the size of neighborhood for the second average filter. The neighborhood size for the first texton map is 3. The size of feature vector now is 320 dimensions (see Figure 5.9).

Table 4 shows that a neighborhood of 6x6 pixels provides the best classification rate.

Neighborhood size	Classification rate
4	92.16
5	92.5
6	**92.83**
7	91.83
8	91.5

Table 4 Neighborhood size for the second average filter

5.10 Comparison

The experimental setup is same as in [48]. We present a comparative evaluation of our approach with five state-of-the-art texture classification methods: [44;46-48;56]. We compare our results with those already published and reported by [48]. we randomly select 100 different training/test splits and report the average classification accuracy, together with the standard deviation, over the 100 runs. The training and test size for each database are reported in Table 5 The choice of parameters for all our experiments on the four dataset are chosen from section

3.2. We use 80 universal textons and the feature vector is of 320 dimensions. Table 6 shows the mean and standard deviations of recognition rate for the compared methods over 100 trials.

Set size Images/ Class	UIUCTex (40)	KTH-TIPS (81)	Brodatz (9)	CUReT (92)
Training	20	40	3	43
Test	20	41	6	49

Table 5 Training and Test set size used for different database.

Database	UIUCTex	KTH-TIPS	Brodatz	CUReT
Ours	92.9 ± 1.2	**97.7 ± 0.8**	92.3 ± 1.0	97.0 ± 0.4
Zhang	**98.3 ± 0.5**	95.5 ± 1.3	**95.4 ± 0.3**	95.3 ± 0.4
Hayman	92.0 ± 1.3	94.8 ± 1.2	95.0 ± 0.8	**98.6 ± 0.2**
VZ-joint	78.4 ± 0.9	92.4 ± 1.4	92.9 ± 1.0	96.0 ± 0.7
Lazebnik	96.4 ± 2.0	91.3 ± 2.1	89.8 ± 0.8	72.5 ± 0.4
G. Gabor	65.2 ± 2.0	90.0 ± 2.0	87.9 ± 1.0	92.4 ± 0.5

Table 6 Comparison of different methods over the 4 database.

5.11 Discussion

Our results show that texture classification can be done with only 3 filters. Previous filter based methods used a bank with 38 filters [41;46] or with 24 filters [56]. We used only 80 universal textons, while [46] used 2440 textons for the CUReT dataset. A comparison of the dimension of the textons for the various methods is shown in

Table 7.

Method	Texton Dimension
Ours	**3**
Zhang SIFT/SPIN	128 / 100
VZ-joint (7x7)	49
Hayman	8

Table 7. Texton dimensions for different texture classification methods

The use of a small size filter preserves the fine details. The new method for selecting the universal textons not only eliminates the learning step, which is performed only once, because we use the same dictionary for all the 4 datasets. The comparison on multiple datasets indicates that texton based methods work with textures which are uniform over the whole image, as in the KTH-TIPS and CURET datasets. The geometrically invariant method proposed by [48] works well also with non-uniform textures. A combination of filter based descriptor and SIFT is expected to improve the classification performance over all datasets. The use of fewer filters, as here proposed, not only saves computation time during classification but also greatly speeds up the learning phase.

6 Neuronal Growth Cone Analysis

Early investigations of the morphology of neurons with AFM [57;58] have allowed a precise measurement of filopodia height but they have also shown that AFM imaging is invasive and that the tip of the used cantilever can modify and distort the membrane of imaged neurons. Indeed when a high force, – in the order of 5–30 nN – is used the tip of the cantilever can damage neurons. This side effect has been utilized to perform nanosurgery, i.e., controlled lesions with a nanoscale dimension. By functionalizing the cantilever tip with appropriate molecules, such as Nerve Growth Factor [59] or antibodies [60], it is possible to localize specific molecules on the surface of neuron and to obtain a high-resolution map of their localization. Confocal laser scanning microscopy (CLSM) utilizes the optical pathway of a regular optical microscope and is capable of collecting three-dimensional (3D) images. [61] and [62] have emphasized that difference in image collection modes between AFM and CLSM make any image comparison a very difficult task and have pointed out the need for an effective integration of these techniques.

We provide a simple and efficient way to integrate confocal and AFM imaging of fixed neurons. Due to the different format of the images acquired by different imaging devices, it is not obvious how to superimpose them. This problem has been extensively addressed in Computer Vision particularly in medical imaging where it is referred to as “Registration”, and solved at some extent [63-66]. We show how to register AFM and confocal images in a coherent and friendly way.

The cellular membrane separates and isolates internal organelles and the nucleus from the external environment, while allowing specific ions and compounds to permeate through it [67]. Its mechanical properties determine the way in which the membrane wraps the cell interior: when the cellular membrane is rigid the cell interior will be wrapped as an inflated balloon and the external surface will be minimal. In contrast, if the membrane is flexible then the cellular membrane will have several invaginations and its total surface will be larger than when it is rigid. Mechanical properties of cells and neurons depend on the elastic properties of the membrane, usually characterized by its Young modulus [68] and by its interactions with the cytoskeleton [69], primarily composed of actin filaments and microtubules. Measurements of the Young’ modulus obtained with the cantilever tip of Atomic Force Microscope in cells provide values ranging from 0.1 up to 400 kPa, i.e. values varying by 3 orders of magnitude [70]. This large variability can be ascribed - to some extent - to different experimental procedures and also to an intrinsic variability of mechanical properties of cells and neurons, which could change significantly in different phases of their cycle and according to the functions which have to be performed. For instance, it is expected that cellular mechanical properties change when cells or neurons move and migrate and when they retract or undergo pruning or other endocytotic processes. Indeed, it has been reported that migrating cells can leave tracks of a small size in the order of 50-100 nm [71]. Similarly, growth cones of axons entering the “decision zone” can pause and holes can appear in their lamellipodium [72;73].

We have investigated the shape and degree of invagination of the external surface of differentiating neurons from dorsal root ganglia (DRG) after 20-30 hours of culture. We have combined Laser Scanning Confocal Microscopy (LSCM) imaging of cytoskeletal components stained with antibodies to high resolution imaging of the external surface with AFM (Kondra et al., 2009). Many growth cones had a fragmented shape composed of several nanoscale microstructures. Such studies can lead to a better understanding of neurodegeneration where the mechanical and morphological properties of neurons and of growth cones change and these changes have a paramount importance.

6.1 Materials and methods

6.1.1 Atomic Force Microscopy

Atomic Force Microscopy (AFM) can provide a precise measurement with a nanometer resolution of the shape - size, height and degree of invagination - of cellular membranes and in particular of growth cones and its constituents, i.e. lamellipodia and filopodia. The advantages of using AFM lie in the nanometer resolution that it offers. Using this technique, we can obtain precise information of the shape, size, and height of a biological sample. The process by which AFM obtains the topographical information of the sample under investigation is via the interaction i.e., the repulsion or attraction between the probe and the sample. The probe is typically a tip at the free end of a silicon cantilever and scans the region of interest. A small laser, focused on the cantilever, detects small deflection of its tip, providing therefore a precise measurement of the distance between the tip and the sample. The result is an image of interaction values providing a three-dimensional (3D) map of the sample height. This information provides a precise characterization of the surface topography of the sample under investigation. The block diagram of AFM is shown in Figure 6.1. AFM provides a direct measure of the height of samples under investigation with a resolution of 1 nanometer (nm) [74] and even less in some special conditions [75]. Lateral resolution better than 1 nm has been achieved also on biological samples operating in liquid [76;77]; however, due to enormous complexity of the tip-sample interaction on biological samples, the minimum resolution achievable can vary significantly from case to case.

AFM was performed using a commercial AFM (Nanowizard II, JPK Berlin) combined with an inverted optical microscope (Zeiss Axiovert 200), and a fluorescence set-up (Zeiss X-cite). Neurons fixed on 24 mm diameter glass coverslips were mounted on the AFM liquid cell, filled with PBS solution and mounted onto the AFM microscope. After laser alignment and tip calibration, the system was left to settle for 30 minutes with laser and microscope condenser switched on to minimize thermal and force drift during image acquisition. AFM was operated in contact mode in liquid medium and the contact force of the cantilever tip was adjusted during imaging to minimize the force exerted by the tip on the sample during scanning. Soft tips from VEECO with low force constant (OBL, 0.03 N/m) were utilized and forces were kept between 100 pN and 1 nN during scanning. AFM and optical images were superimposed using commercial Software (DirectOverlay, JPK, Berlin) that corrects the optical aberrations using the AFM scanner position sensors with an accuracy of 2-3 nm.

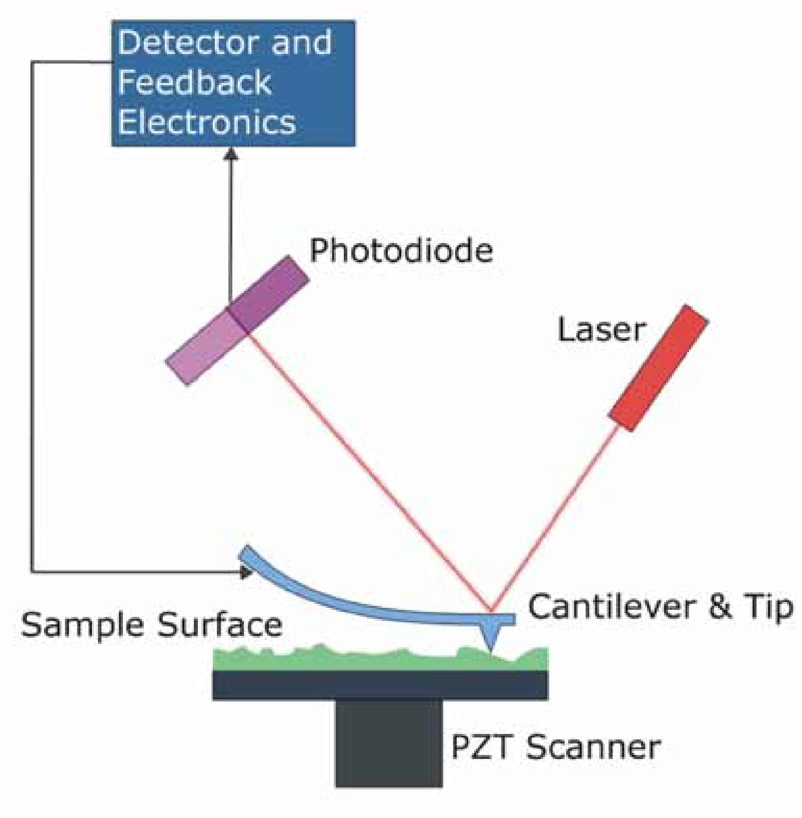


Figure 6.1 Block Diagram of Atomic Force Microscope

6.1.2 Dorsal root ganglia isolation and culture

Dorsal root ganglia (DRG) were prepared from P10–12 Wistar rats as previously described [78]. Wistar rats (P10–12) were anesthetized with CO_2 and sacrificed by decapitation in accordance with the European Communities Council Directive of 24 November 1986 (86/609/EEC) and the Italian Animal Welfare Act and approved by the Local Authority Veterinary Service. DRGs were incubated with 0.5 mg/ml trypsin, 1 mg/ml collagenase, and 0.1 mg/ml DNase (all from Sigma) in 5 ml Neurobasal medium (Invitrogen) in a shaking bath at 37 °C for 35–40 min. They were mechanically dissociated, centrifuged at 300 rpm, resuspended in culture medium, and plated on 0.5 μg/ml poly-l-lysine (Sigma) and Matrigel (BD Biosciences, Bedford, MA, USA)-coated 24 mm coverslips in Neurobasal medium containing 10% fetal FBS (Sigma) and 100 ng/ml nerve growth factor (NGF, Alomone, Israel) at a density of 2.5×10^5 cells/cm^2. Cells were then fixed 20-30 hours after differentiation.

6.1.3 Cell Fixation

Before imaging with LSCM and AFM, DRG neurons were fixed. In order to verify that the observed structural details were not caused by the specific fixation method, two fixation procedures were used and compared: with Paraformaldehyde and Glutaraldehyde.

Fixation with Paraformaldehyde (PFA): DRG neurons were fixed by incubating in 4% PFA (Sigma-Aldrich) containing 0.15% picric acid in phosphate-buffered saline (PBS) for 20 minutes at room temperature (RT), rinsed in PBS, and incubated in blocking solution (consisting of 5% fetal bovine serum, 0.3% Triton X-100 in PBS) for 30 minutes at 37°C.

Fixation with Glutaraldehyde (GLU): DRG neurons were perfused 3 times with 0.25% GLU (Sigma-Aldrich) in buffer solution (consisting of 100 mM Cacodylate (pH 7.4), 5 mM CaCl2 and 10 mM MgCl2) and incubated in buffer solution for 10 minutes at 37°C (Rochlin et al., 1995).

6.1.4 Confocal Imaging

Figure 6.2 shows the principle for confocal microscopy. CLSM using fluorescence mode can collect emissions originating from the interior of a biological sample and therefore CLSM has been widely used to map intracellular mechanisms. When antibodies are used to stain specific biological structures, fluorescent confocal microscopy provides a powerful way to simultaneously map the distribution of different cellular components [61;79]. The 200 nm resolution limit of the CLSM is restricted by the diffraction limit of the microscope objective and does not permit accurate evaluation of the sample height. The z-resolution of CLSM for any given wavelength is always at least 2 times less than the corresponding xy-resolution.

Cells were imaged using a confocal (Leica DMIR2) microscope, equipped with Diode laser (emitting at 405 nm), Ar/ArKr (at 488 nm), He/Ne (at 543/594 nm) and He/Ne (at 633 nm) lasers. A stack of 4-7 images, 400 to 700 nm apart, were collected with a 40X magnification and 1.44 NA oil- immersion objective. The averaged image of the stack was used to compare with the images acquired using AFM.

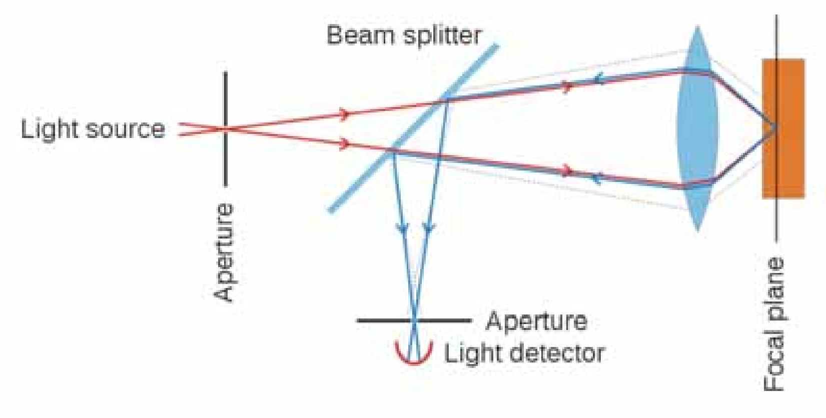


Figure 6.2 Principle of confocal microscopy

6.2 Image Registration

AFM and CLSM images contain different information: every pixel at location $(x,y)_{AFM}$ in the AFM images provides a direct measurement in nm of the sample height while fluorescence images acquired with a confocal laser scanning microscope (CLSM) characterize the emitted fluorescence at location $(x,y)_{CLSM}$. In order to integrate these information it is important to properly align or superimpose these different images. This problem has been extensively studied in Computer Vision, where it is referred as "Registration" of different images, and it will be addressed in the next section.

Image registration is the process of overlaying two or more images of the same scene taken at different times, different viewpoints, and/or by different imaging modalities. Differences between images are introduced because of different imaging conditions, such as different viewing points or light conditions or because of the use of two different microscopes, as in the case under consideration. Registration transforms one image – usually the sensed image – so that it becomes aligned to the reference image. Image registration is a crucial step in image analysis and can be solved using methods used in computer vision [5], in which the final information is obtained by integrating various data sources like in image fusion [80]. Images taken from different modalities may undergo a linear deformation in scale, translation, rotation and sometimes shearing i.e. affine deformation, the transformation is in general non-linear.

If (x,y) is the system of reference of image A and (x',y') the system of reference of image B. The transformation T is defined as:

$$\begin{bmatrix} a1 & a2 & t_x \\ a3 & a4 & t_y \\ c1 & c2 & 1 \end{bmatrix} \qquad \textbf{eq. 5}$$

Where:

$\begin{bmatrix} a1 & a2 \\ a3 & a4 \end{bmatrix}$ is a rotation matrix. This matrix defines the kind of the transformation that will be performed: scaling, rotation, and so on.

$\begin{bmatrix} t_x \\ t_y \end{bmatrix}$ is the translation vector. It simply moves the points.

$\begin{bmatrix} c1 & c2 \end{bmatrix}$ is the projection vector.

If x and y are the coordinates of a point, the transformation can be done by the simple multiplication:

$$\begin{bmatrix} a1 & a2 & t_x \\ a3 & a4 & t_y \\ c1 & c2 & 1 \end{bmatrix} \times \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} \qquad \textbf{eq. 6}$$

Here, x' and y' are the coordinates of the transformed point.

For Linear transformations

c1 and c2 are equal to 0.

$\begin{bmatrix} a1 & a2 \\ a3 & a4 \end{bmatrix}$ can be written as $\begin{bmatrix} sc & -ss \\ ss & sc \end{bmatrix}$

Where sc = scale * cos(θ), ss = scale * sin(θ), and θ is the rotation.

Thus the transformation matrix becomes

$$T = \begin{bmatrix} sc & -ss & t_x \\ ss & sc & t_y \\ 0 & 0 & 1 \end{bmatrix} \qquad \textbf{eq. 7}$$

At least two pairs of corresponding points are needed to solve for the four unknown coefficients (sc, ss, , tx, ty)

For Affine transformations

c1 and c2 are equal to 0.

Thus the transformation matrix becomes

$$T = \begin{bmatrix} a1 & a2 & t_x \\ a3 & a4 & t_y \\ 0 & 0 & 1 \end{bmatrix} \qquad \textbf{eq. 8}$$

At least three pairs of corresponding points are needed to solve for the six unknown coefficients.

For projective transformations, The transformation matrix is

$$T = \begin{bmatrix} a1 & a2 & t_x \\ a3 & a4 & t_y \\ c1 & c2 & 1 \end{bmatrix} \qquad \textbf{eq. 9}$$

At least four pairs of corresponding points are needed to solve for the eight unknown coefficients.

The word registration is used with two different meanings [63]. The first meaning is to determine a transformation of one image so that features in the sensed image can be put in a one-to-one correspondence to features in the reference image. The symbol T is used to represent this type of transformation. The second meaning of registration enables also the comparison of the intensity at corresponding positions. The symbol T_i is used to describe this second meaning of registration, which incorporates the concepts of re-sampling and interpolation. Using the language of geometry, the registration transformation is referred to as the mapping T, transforming a position $(x,y)_A$ in the system of reference of image A to the position $(x,y)_B$ in the system of reference of image B.

$$\mathbf{T : (x,y)_A \rightarrow (x,y)_B \Leftrightarrow T\ (x,y)_A = (x,y)_B} \qquad \textbf{eq. 10}$$

Using this notation, T is a spatial mapping. The more complete mapping T_i maps both position and associated intensity value from image A to image B. Therefore T_i maps an image to an image, whereas T maps between coordinates of image A into coordinates of image B. To overlay two images that have been registered, or to subtract one from another, is necessary to know T_i, not just T. T_i is only defined in the region of overlap of the image fields of view, and has to take into account of image sampling and spatial resolution.

Before doing the registration, it is necessary to select the correct type of transformation required for the images under consideration. The linear transformation is used when shapes in the sensed image are unchanged, but are distorted by some combination of translation, rotation, and scaling (Figure 6.3). Rectangles remain rectangles. This is the case of the same biological sample viewed by two microscopes using different objectives and imaging systems, such as an AFM (Figure 6.4 left panel) and a CLSM (Figure 6.4 right panel). Registration using affine transformation is necessary when shapes in the sensed image are distorted also by shearing or a linear deformation (Figure 6.3). In this case, straight lines remain straight, and parallel lines remain parallel, but rectangles become parallelograms. Projective transformation is used when the scene appears tilted (Figure 6.3). Straight lines remain straight, but parallel lines converge toward vanishing points that might or might not fall within the image.

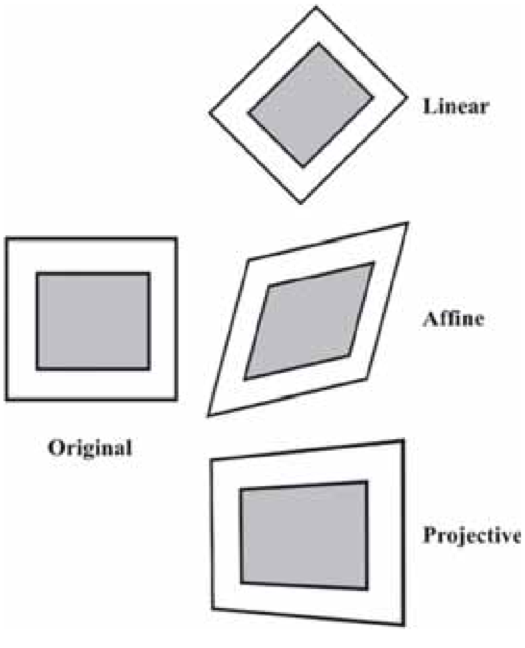


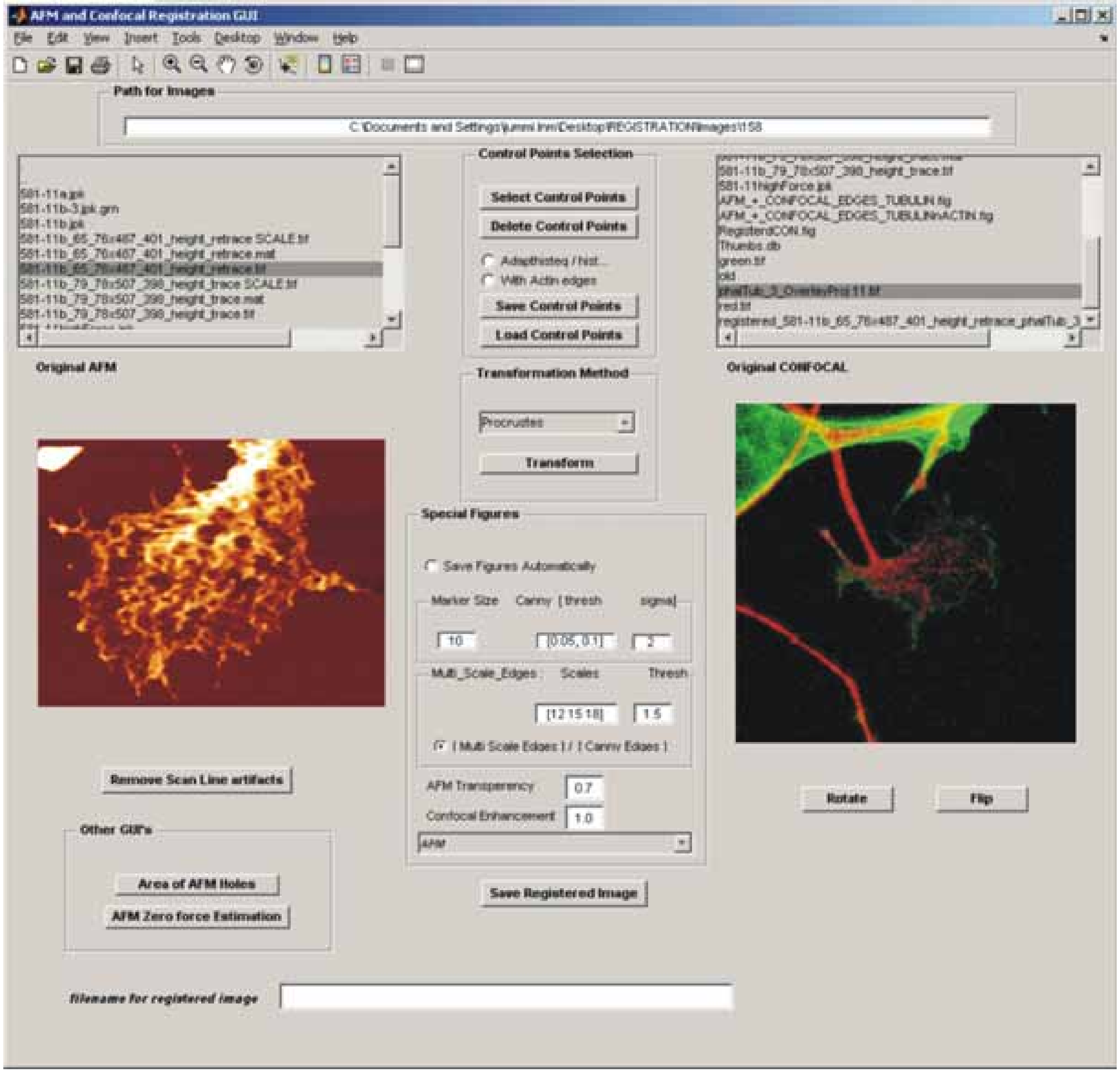
Figure 6.3 Examples of 2D transformations

Let us now consider two images of the same biological sample S, the first one acquired using AFM ($I_{AFM}(x',y')$) and the second using a CLSM ($I_{CLSM}(x,y)$). The two imaging systems differ because of different scale factors, by a rigid translation of their origin (t_x,t_y) and a possible rotation θ of their axis. It is assumed that the relation between the images is affine so that the transformation between the two systems of coordinates (x,y) and (x′,y′) can be solved using eq. 7or eq. 8. Registration of the two $I_{AFM}(x',y')$ and $I_{CLSM}(x,y)$ consists in the determination of the transformation T (see eq.(10)) by aligning properly two images such that $I_{AFM}(T(xi',yi'))$ and $I_{CLSM}(xi,yi)$ is the AFM and CLSM of the same physical point.

The unknown parameters in the transformation matrix T can be estimated either by matching a selected number of points or landmarks in the two images (6.3.1) or by matching entire contours in the two images (6.3.2). The first method is used when it is possible to identify in the two images $I_{AFM}(x',y')$ and $I_{CLSM}(x,y)$ enough points corresponding to the same physical structure, such as the tip of a dendrite, or a small vesicle. The second method is used in presence of rounded biological structures, with no obvious marks and it is more convenient to put in correspondence two contours and not isolated points.

To apply these methods to the images derived from different microscopes we have implemented our Registration program using Matlab7.1 (The MathWorks Inc. http://www.mathworks.com) a standard tool common in the computer vision and biomedical community for statistical analysis and production of figures and images of analyzed data and results.

Corresponding points or contours in $I_{AFM}(x',y')$ and $I_{CLSM}(x,y)$ were marked by hand using a friendly graphical user interface developed for this particular purpose see Figure 6.4



Path for Images: C:\Documents and Settings\jummi lnn\Desktop\REGISTRATION\images\158

Transformation Method: Procrustes

Marker Size: 10
Canny [thresh: [0.05, 0.1]
sigma]: 2

Multi_Scale_Edges Scales: [12 15 18]
Thresh: 1.5

● [Multi Scale Edges] / [Canny Edges]

AFM Transperency: 0.7
Confocal Enhancement: 1.0
AFM

Figure 6.4 GUI for Image registration

6.2.1 Registration by points selection method

This method is illustrated in Figure 6.5. The operator identifies N (minimum 3) points in the $I_{AFM}(x',y')$ which are put in correspondence with N points in the $I_{CLSM}(x,y)$. Corresponding points in Figure 6.5A and B are indicated by the same number. By using these correspondences the parameters determining the transformation T are obtained by solving a system of linear equation and the $I_{CLSM}(x,y)$ can be registered. In this way same physical points in $I_{AFM}(x',y')$ and in the new registered image $I^*_{CLSM}(x,y)$, shown in Figure 4C, have the same location (x,y). To make the selection of points more user friendly, $I_{CLSM}(x,y)$ was rotated in a sequence of 90 degrees and/or flipped to make a approximate alignment with $I_{AFM}(x',y')$. Control Point Selection Tool in Matlab was used to mark the control point pairs in the image to be registered, the input image Figure 6.5B, and the image to which you are comparing it, the base image (Figure 6.5A) corresponding points were initially specified by pointing and clicking in the input and base images so that each point specified in the input image had a match in the base image.

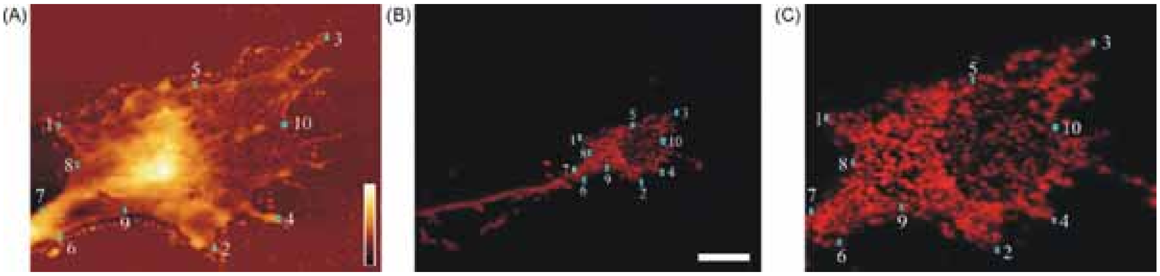


Figure 6.5 Example of registering images with point selection method
(A) AFM image of ES-derived growth cone. Color bar from 0 to 350 nm. (B) Confocal image of the same ES-derived growth cone. Scale bar = 10 µm. (C) Registered confocal image.

The advantage of this tool is that, after the first three pairs are selected, it is sufficient to select a point in the input image and the Control Point Selection Tool estimates its match point in the base image or *vice versa* automatically, based on the geometric relationship of the previously selected control points. Another advantage of this tool is that if the image is not registered properly, it is possible to change the position of control points to get the exact superimposition.

6.2.2 Registration by aligning contours with Procrustes Analysis

Procrustes was a robber in Greek mythology. He would offer travelers hospitality in his road-side house, and the opportunity to stay the night in his bed that would perfectly fit each visitor. As the visitors discovered to their cost, however, it was the guest who was altered to fit the bed, rather than the bed that was altered to fit the guest [63]. To make the idea more clear let us consider that the AFM image, $I_{AFM}(x',y')$ is like the bed and $I_{CLSM}(x,y)$ is the guest. So using the Procrustes' method means that $I_{CLSM}(x,y)$ is altered by scaling and/or rotating to approximately fit $I_{AFM}(x',y')$.

The number of landmarks available depends on the structure of the image, and there may be differences in opinion between scientists as to which landmarks are consistent and locatable. Marking of the corresponding points thus becomes difficult when there are few corners in the structure, and thus exact location of the points is impossible and can differ from observer to observer. This problem can be solved by marking many points along the contour of the structure in both images. The contours are then interpolated so that both of them contain the same number of equally spaced points.

Given two contours S_1 and S_2 with the same number of points N, the Procrustes analysis finds the best transformation T of one contour, i.e. $S_2^* = T(S_2)$ in such a way that the distance between the two contours S_1 and S_2^*

is minimal [81]: It involves least square matching of two configurations using the similarity transformations. The parameters are estimated using the squared Euclidean distance.

$$\left\| S_2^* - S_1 \right\|^2 \qquad \textbf{eq. 11}$$

By restricting transformation to the similarity case, we have:

$$T\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{bmatrix} a & -b & tx \\ b & a & ty \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \qquad \textbf{eq. 12}$$

If the two shapes have the same center of gravity located in the origin, the optimal transformation is:

$((t_x, t_y) = (0,0))$,

$$a = \left(\sum_{i=1}^{n} (x_i x_i' + y_i y_i') \right) / \left\| S_1 \right\|^2, \quad b = \left(\sum_{i=1}^{n} (x_i y_i' - y_i x_i') \right) / \left\| S_1 \right\|^2 \qquad \textbf{eq. 13}$$

Where $S_1 = (x_i, y_i)$ and $S_2 = (x_i', y_i')$. $\left\| S_1 \right\|$ is the Euclidean norm.

$$\textbf{Scale} = \sqrt{a^2 + b^2} \qquad \textbf{eq. 14}$$

$$(t_x, t_y) = (\overline{xi}, \overline{yi}) \qquad \textbf{eq. 15}$$

Where $\overline{xi}, \overline{yi}$ are mean of the x and y coordinates of the contour S_1

Here the translation was removed only for the second contour. The first contour S_1 was fixed like the bed of Procrustes. As we did not have the optimal correspondence for the two point-sets, it was necessary to consider N possible cyclic renumbering iteratively for each shape, so that the optimal alignment was obtained when the full Procrustes distance (eq. 11) was minimal. We use Procrustes analysis only with rotation and translation and not Affine. The scale can be recovered using eq.14.

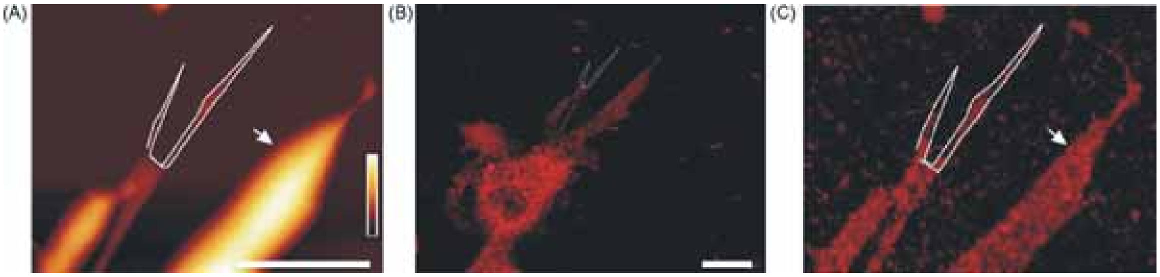


Figure 6.6 Example of registering images with Procrustes analysis

(A) AFM image of ES-derived growth cone. Color bar from 0 to 2200 nm. Scale bar = 10 µm. (B) Confocal image of the same ES-derived growth cone. Scale bar = 10 µm. (C) Registered confocal image of membrane protein NCAM.

An example of this method is illustrated in the Figure 6.6. A contour was marked by continuously clicking on points following the borderline of the structure of interest in I_{AFM} (Figure 6.6A) and in I_{CLSM} as well (Figure 6.6B). As a consequence of Procrustes analysis, $I_{CLSM}(x,y)$ was properly transformed and put in correspondence with $I_{AFM}(x',y')$, as shown in Figure 6.6C.

When $I_{AFM}(x',y')$ and $I^*_{CLSM}(x,y)$ have been appropriately registered, they can be superimposed to localize the labeled proteins with respect to cell topography. In the following examples when many prominent corners were present in the growth cones, the CLSM images were registered by points selection method (Figure 6.7A and B). In such cases Procrustes analysis was not used because marking a contour was very laborious. This method was applied in cases where the growth cone had a smoother contour (Figure 6.7C and D). Superimposition can be achieved in different ways. Registered Confocal images can be overlayed with the corresponding AFM image as shown in Figure 6.7.

Image overlay is a promising data visualization technique with a wide variety of medical applications. The 3D image of AFM was first plotted and then the confocal image was texture mapped on a flat surface below it as shown in Figure 6.7 column 3. An alternative way to visualize the information from these two types of images is to texture map the confocal image on the 3D AFM image as shown in Figure 6.7 column 4. In both cases the 3D image can be rotated and/or zoomed to obtain a better view of the structure.

The 3D overlay gives an indicative correspondence of the presence of sub-cellular components marked with fluorescence with topographical information from AFM. This way one can get a quick, accurate and detailed understanding of the structure before doing any further analysis such as calculating average height of a particularly stained protein.

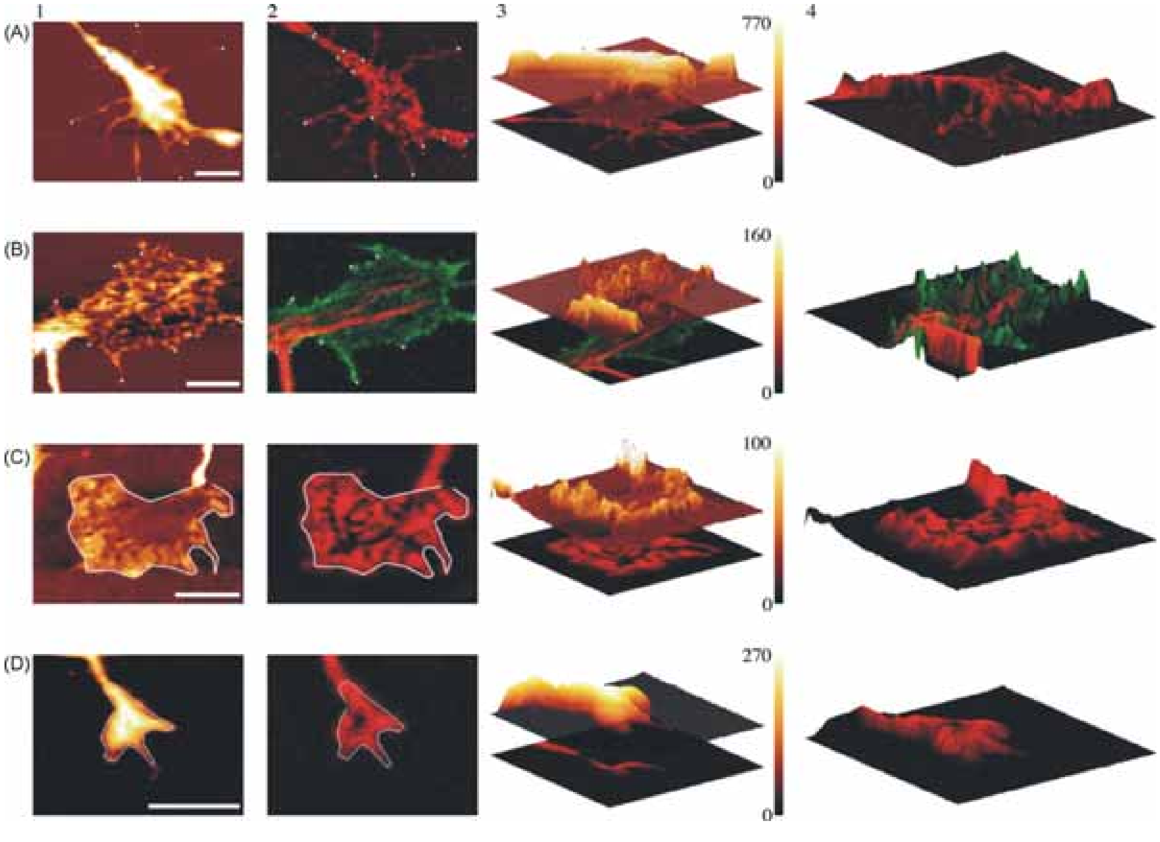


Figure 6.7 Registered Confocal images overlayed with the corresponding AFM image

Points selection method (indicated with white dots) was used in rows A and B while Procrustes analysis was applied for images in rows C and D (indicated with white lines). (Column 1) AFM images of ES-derived (A, C, D) and DRG neuron (B). Scale bar = 5 μm. (Column 2) Confocal Images of the same neurons in column 1. (Column 3) 3D view of the overlayed images. Color bar represents values in nm. (Column 4) Confocal images texture mapped on the 3D AFM images.

6.3 Zero Force AFM Image

The operation of an AFM requires the application of a force to its cantilever. The minimal force required to acquire a stable height map of a sample is of the order of 100 pN. In order to verify whether properties of growth cones measured with AFM are affected by the force applied by the AFM cantilever on the sample, we obtained a series of AFM images of the same growth cone at increasing scanning forces, from 100 to 2000 pN, and then repeated with the lowest scanning force to confirm that the growth cone has not been destroyed by the multiple scans. As shown in Figure 6.8, very similar AFM images were obtained with low (100 pN) and high (2000 pN) scanning forces and the morphology is not destroyed. But it must be pointed out that higher scanning forces tend to press the more flexible structures evident from the scan lines.

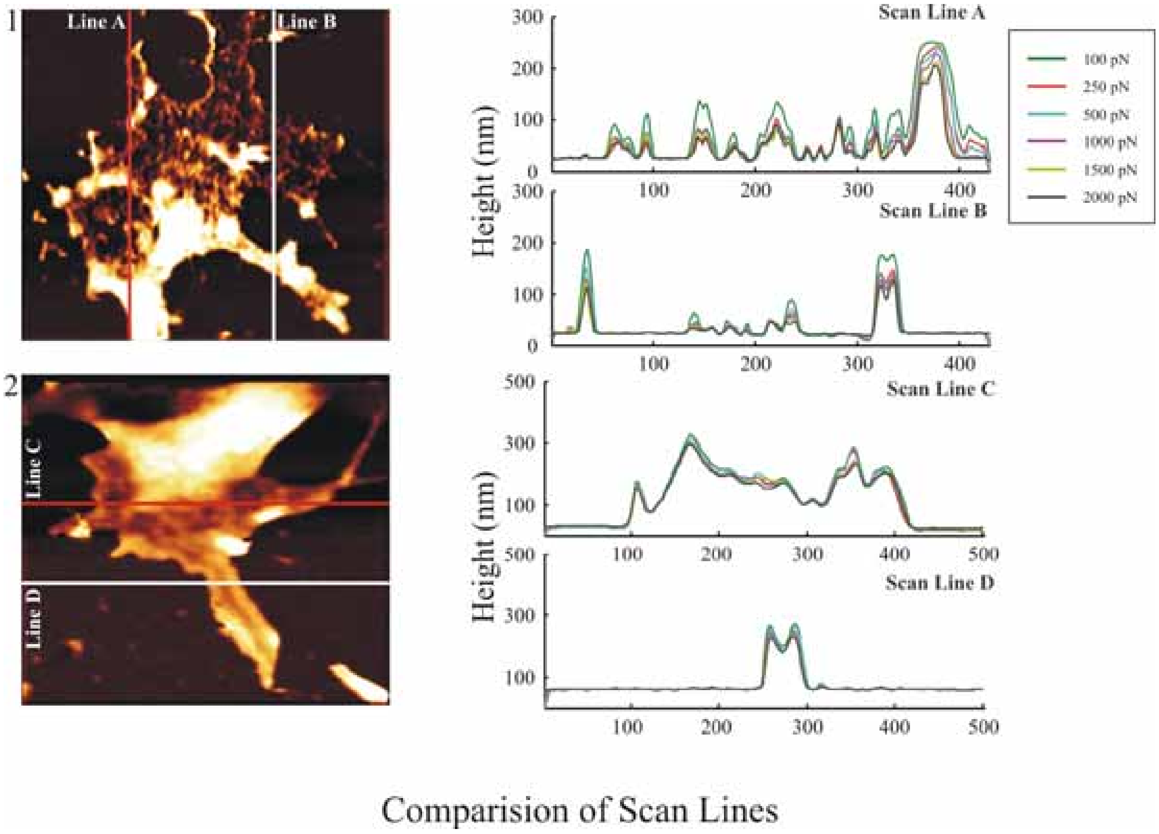


Figure 6.8 Need for a zero force Image
line profiles of the scan lines shown in 1 and 2.

In order to eliminate this problem, we developed a procedure to obtain *0 pN AFM Images* by extrapolation. Once the AFM images are obtained at scanning forces varying from 100 to 2000 pN, they are registered with the image taken at the lowest force. The AFM image at *0 pN AFM Image* is obtained using *spline* interpolation of the pixel intensities obtained at different forces. The procedure is as follows :-

1. AFM Images for growth cone ‘A’ at forces ranging from 100 pN to 2000 pN are obtained.
2. AFM images are registered and a stack is assembled from all the registered images (Figure 6.9A).
3. Using the stack of growth cone ‘A’, we obtain two images (Figure 6.9C) one extrapolated at 0 pN and another at 390 pN (force used for scanning another growth cone B see (Figure 6.9B)) using *interp3* command in Matlab (*spline* interpolation of the pixel intensities obtained at different forces is used internally in this function).

4. Using these two images, we obtain a 4th degree polynomial function which maps the height values at 390 pN to the one at zero force for growth cone 'A' (Figure 6.9D).
5. Using this function, the *0 pN AFM Image* for growth cone 'B' is obtained (Figure 6.9E).

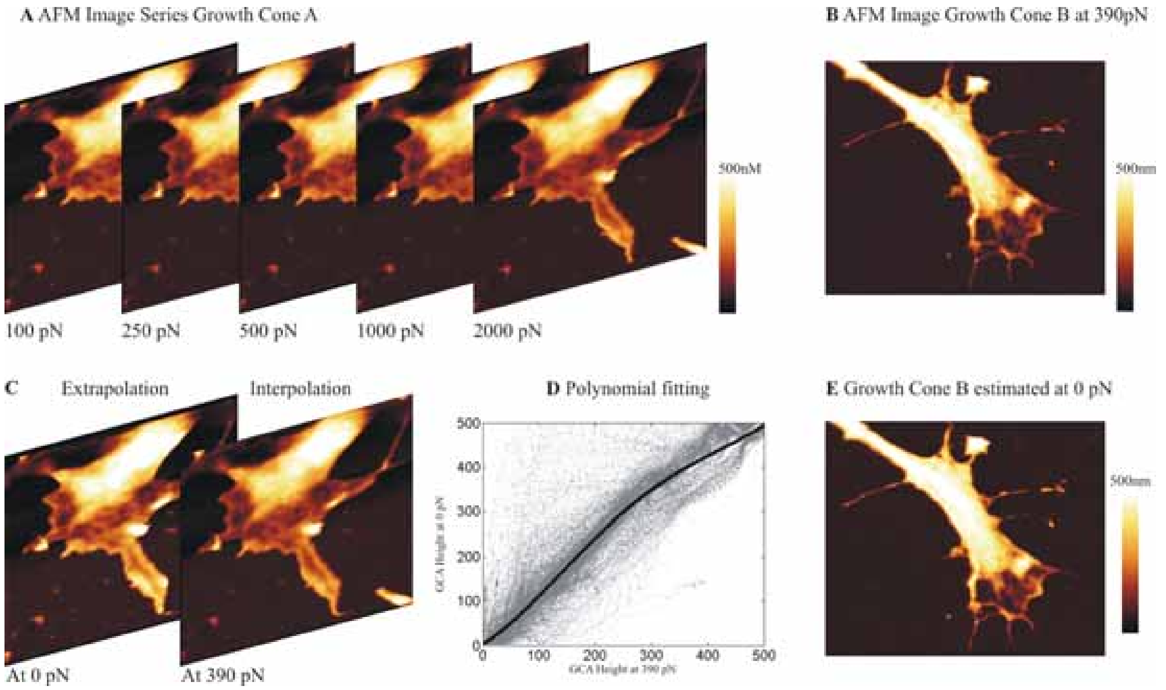


Figure 6.9 Steps for obtaining Zero force AFM Image

This procedure can also be used to obtain the height map at any desired scanning force. In that case, one should interpolate between the available scanning force and the desired one (at step 3).

6.4 Morphological Analysis

6.4.1 Selecting region of interest

The outer membrane of DRG growth cones when imaged with AFM often exhibit significant invaginations and holes. We have characterized these holes by measuring the depth and size of these holes by using tools inspired by Computer Vision, such as snake algorithms [82;83]. An approximating closed region Γ around the growth cone is selected by the user (Figure 6.10B green trace), and then a snake algorithm is applied, which drives Γ to the boundary of the growth cone. (Figure 6.10B blue trace). The surface of an arbitrary surface can be approximated by Delaunay triangulation [84]. Holes are detected and analyses were performed following the protocol described in below by using a graphical interface written in Matlab 7.1 (The MathWorks Inc. http://www.mathworks.com).

To determine the holes, we use the following algorithm implemented in MATLAB 7.1:

1. Display AFM image where the intensity corresponds to height.
2. Choose threshold Th. Th is usually set to 15% of the maximum intensity value of the image.
3. Define a region R of the AFM image as a hole if the following conditions are fulfilled:

   a. Intensity averaged over R (Int(R)) is less than Th. As AFM images intensity corresponds to height, this procedure is equivalent to set a threshold on its average height,
   b. Area of R is less than 5% of the total area of the base of the growth cone.
   c. R is completely surrounded by a non-hole (hilly) region.

   False positives are eliminated using these conditions. True holes are shown in Figure 6.10C.

4. Repeat Step 2 and 3 with increasing values of Th till no holes can be found.
5. Merge all the holes found with all the different values of the threshold.
6. Calculate base area. Scale to real dimension (μm2 in our case).
7. Calculate height of the 'hill' as the height of a sector of width of 5 pixels surrounding the hole. This gives the depth of the hole. Scale to real dimension (nm in our case).
8. Calculate height of the hole as the difference between the mean height of the 'hill' surrounding the hole and the mean height of the hole i.e., the threshold used to find the holes.
9. Perform triangulation of surface/height given by the AFM image and find area of each triangle. The sum gives total area of the surface. Scale to real dimension (μm2 in our case).
10. Ratio of total area to base area indicates the texture of the surface: if close to 1, surface is flat and devoid of holes, whereas higher values indicate a surface with many holes.
11. Determine other parameters as required.

If R has been classified as a hole, we consider a sector S of width of 5 pixels surrounding R and the depth or height of the hole R is defined as Int(S)- Int(R). Hole area is calculated as the area of the region R, scaled to real dimensions.

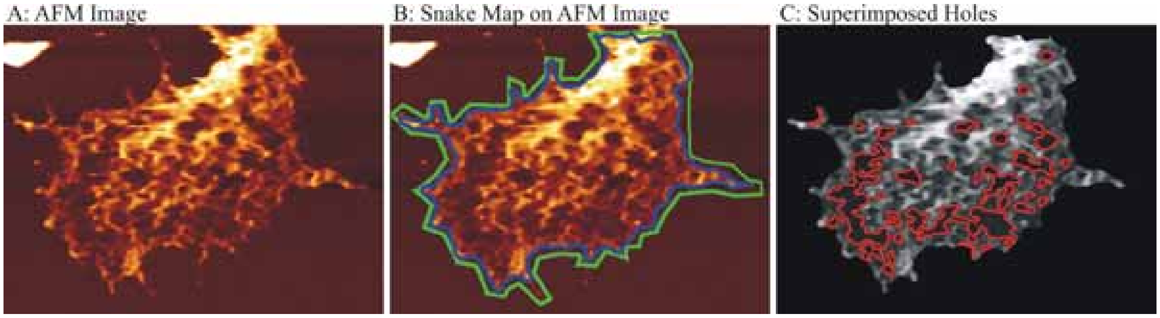


Figure 6.10 Selection of growth cone and finding AFM Holes

A: Original AFM image. B: Selection of growth cone for calculation by Snake Map. C: holes found after calculation (i.e., below threshold)

For each growth cone, we measured:

the *total area* (T), defined as area of the entire growth cone, including the hole walls and various invaginations (i.e., sum of area of the triangles representing the surface approximated by Delaunay triangulation);

the *base area* (B), defined as the area of the growth cone projected on the XY-plane;

the *invagination value* I, a parameter defined as: $I = \frac{T}{B \cdot H_{max}}$, where H_{max} is the maximum height of the growth cone.

If the growth cone is more invaginated, it has more folds on the surface and thus this ratio is high. Collected data from the 13 analyzed growth cones are reproduced in Table 8. All AFM images shown in Figure 6.11 are extrapolated images for 0 pN scanning force (see previous section).

6.4.2 Comparison of Fixation Methods

Fixation Method	AFM Image	Total Surface Area (μm^2)	Total Hole Area (μm^2)	Base Area (μm^2)	Total Area/ Base Area	Number of Holes	Maximum Height of Holes (nm)	Invagination Value
PFA	Image-1	599	14	132	4.2	55	200	2.1
	Image-2	136	7	33	4.1	16	140	3
	Image-3	603	51	143	4.2	29	160	2.6
	Image-4	765	108	187	4.1	20	160	2.6
	Image-5	520	46	127	4.1	19	150	2.7
	Image-6	245	10	55	4.5	60	250	1.8
	Average	478±239	39.3±23	112.8±57	4.2±0.16			
GLU	Image-11	1137	30	270	4.2	59	400	1.1
	Image-12	965	144	232	4.2	37	250	1.7
	Image-13	280	17	62	4.5	9	500	0.9
	Image-14	187	9	41	4.6	10	500	0.9
	Image-15	395	62	93	4.3	9	450	1
	Image-16	161	19	37	4.4	70	400	1.1
	Image-17	670	43	159	4.2	16	800	0.5
	Image-18	95	0	19	5	No holes	650	0.8
	Image-19	144	0	32	4.5	No holes	650	0.7
	Average	448.2±285	36.1±23	105±45	4.4±0.3			

Table 8 Comparison of parameters from growth cones fixed using the two different Protocols

In order to understand whether the fixation procedure affects morphological properties of the growth cone, we used two different fixation procedures, one using PFA and the other using GLU. As these two chemicals fix the cell through different processes [85]) it is useful to verify whether detected nanometer dimensional structures are introduced by treatment with these chemicals. We analyzed growth cones with AFM using scanning forces from 100 pN up to 2000 pN so to obtain 0 pN AFM images, representing the outer surface of the growth cone with a minimal distortion introduced by the cantilever tip. The outer surface of 13 over 15 imaged growth cones was not smooth, but instead was highly corrugated and had a significant number of holes.

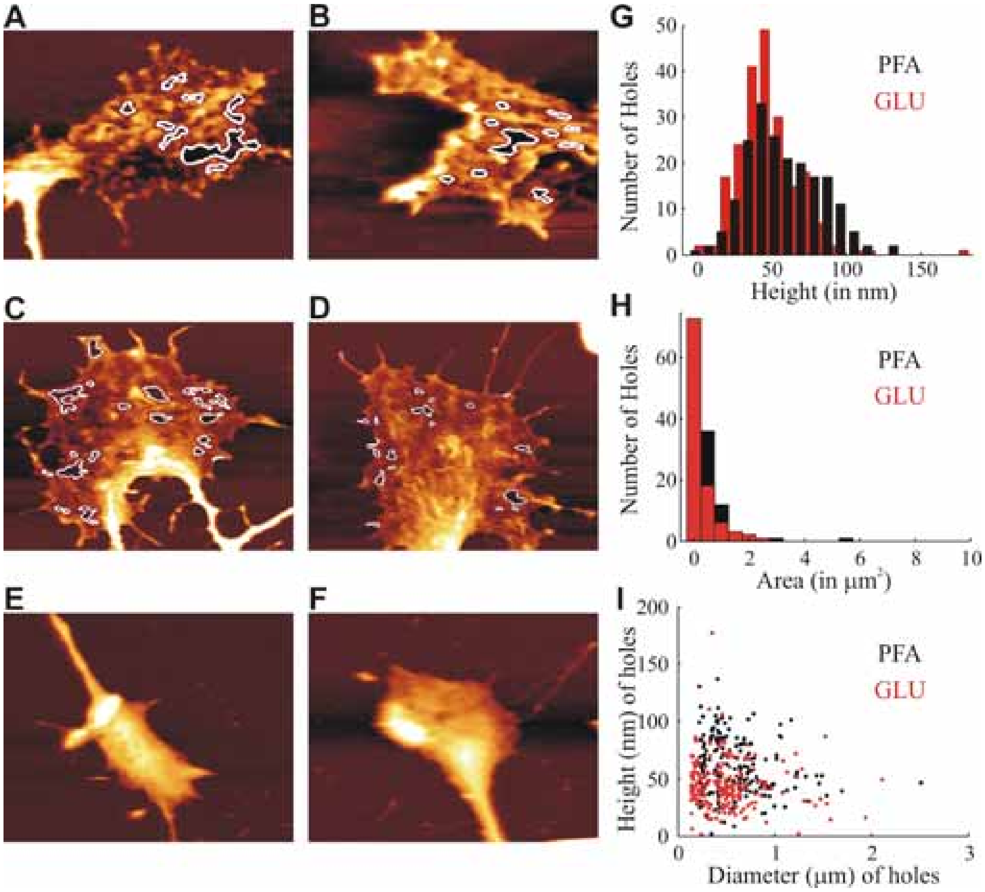


Figure 6.11 Comparison of Fixation Methods

A-D: AFM images (interpolated for 0 pN) after fixing with PFA (A, B) and GLU (C, D). White contours indicate holes. E-F: Examples of growth cones without holes. G-I: Distribution of height (G) and area (H) of holes observed in the growth cones after fixation using PFA and Glutaraldehyde. I: Plot of height against diameter of different fixatives. Black denotes PFA and Red GLU

As shown in Figure 6.11, holes were observed when growth cones were fixed both using PFA (6 growth cones) and GLU (7 growth cones). Having observed an invaginated surface with holes, we attempted to quantify their properties and therefore we measured the height of a hole as the difference between the mean height of the "hill" surrounding the hole and the threshold used to find the holes and other parameters such as their area. The distributions of height and area of detected holes are shown in Figure 6.11G, H respectively and are very similar for both fixation procedures. Properties of holes varied significantly among individual growth cones, as shown in Table 8 and Figure 6.11. The number of detected holes varied from just 9 holes (image-13 and image-15) to 70 holes in image-16. The height of holes varied from just 2 nm up to 178 nm and their average height was about 51 nm (see Table 9). The area of their base was rather varying with a range from 0.01 to 3.5 μm².

Parameters	PFA (n = 6)	GLU (n = 7)	Both fixation protocols (n = 13)
Height Range	2 to 128 nm	2 to 178 nm	2 to 178 nm
Average Height	57 ± 23 nm	45 ± 21 nm	51 ± 23 nm
Base Area of Holes Range	0.04 to 2.3 μm²	0.01 to 3.5 μm²	0.01 to 3.5 μm²
Average Base Area	0.35 ± 0.49 μm²	0.28 ± 0.45 μm²	0.32 ± 0.47 μm²

Table 9 Hole statistics

These results show that growth cones fixed with either GLU or PFA do not always have a smooth external surface but often have holes with nanometric dimensions, which cannot be seen by conventional optical microscopes

As AFM provides a rather accurate 3D representation of imaged growth cones, we characterized several geometrical properties of its surface. Three quantities were computed, i.e. *total area*, *base area* and *hole area*), providing a characterization of the outer surface of growth cones. Collected data from the 15 analyzed growth cones are reproduced in Table 8. All measurements are based on the extrapolated 0 pN AFM images. In order to quantify the smoothness of a surface, we have introduced the parameter, *invagination value,* I, a low value of I (around or less than 1) indicates a smooth external surface and a higher value of this quantity is found for growth cones with many holes having a significant height. The *invagination value* for all tested growth cones fixed with either PFA or GLU, varied from 0.5 to 3.

6.4.3 AFM-CLSM combined analysis

In order to correlate the measured height with the cytoskeletal components of the analyzed growth cones, we applied immunofluorescent techniques where actin is fluorescently label with Alexa488 phalloidin and tubulin with anti-β tubulin III, followed by Alexa594 anti-mouse IgG. This enabled us to visualize the correspondence of the observed invaginations with the internal cytoskeletal structure and in particular with the presence or absence of actin filaments and/or tubulin. LSCM scanning were performed first on the fluorescently labeled growth cone. 4-7 slices of fluorescence images were acquired with a step size of 500 nm. The stack of these images was then averaged and used for further analysis. AFM was then performed on the same growth cone. Acquired AFM and averaged LSCM images were registered as described in [86] Figure 6.12 show regions of the growth cone with a height more than 100 nm exhibiting a significant co-localization with the staining of actin filaments and tubulin, indicating that these proteins act as pillars inside the growth cones. In contrast, holes (area within white contours, Figure 6.12) show poor co-localization with actin filaments and microtubules.

In order to verify whether holes in AFM images correspond to absence of major cytoskeletal components, we binarized fluorescent actin and tubulin images as well as AFM images (green, red and blue regions respectively in Figure 6.12). Using a simple AND operation, we computed the number of pixels which are common to binarized tubulin or binarized actin and the holes. Table 10 shows the result of our analysis of the correspondence of the topography of the growth cone to the presence (or absence) of actin and tubulin through AFM scanning and subsequent confocal imaging.

Parameters	Image-1	Image-3	Images-4	Image-5	Average ± Standard Deviation
Tubulin inside holes (%)	18	2	6	3	7 ± 7
Actin inside holes (%)	13	18	19	15	16 ± 3
Actin + Tubulin outside holes (%)	55	60	64	62	60 ± 4
Tubulin/Actin ratio of entire growth cone	1.47	0.57	1.05	0.67	
Tubulin Height (nm)	146	162	109	127	136 ± 22
Actin Height (nm)	115	112	79	94	101 ± 16
Tubulin Height /Actin Height	1.27	1.44	1.37	1.35	1.36 ± 0.07

Table 10 Comparison of Actin Tubulin Ratio in growth cones

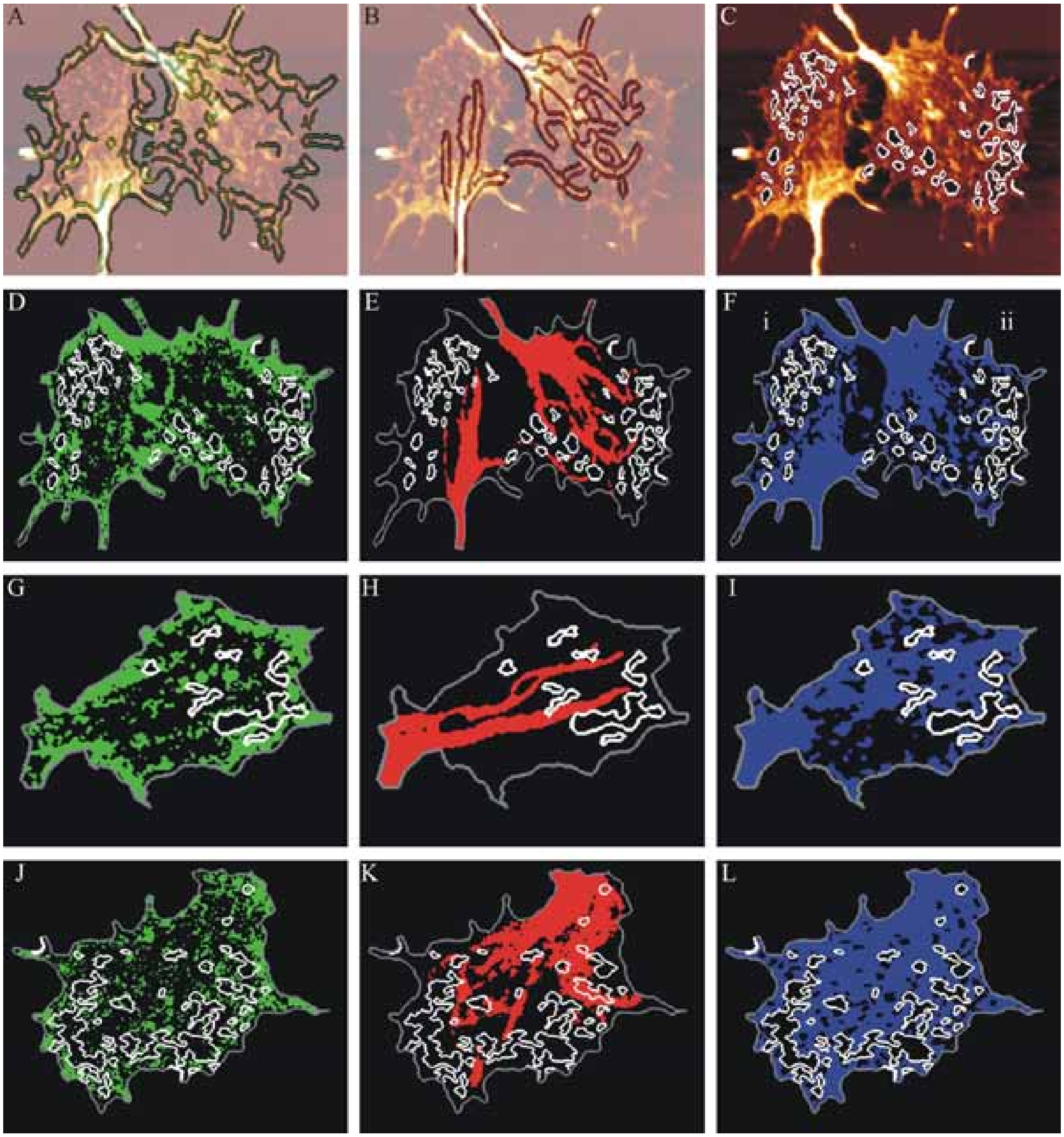


Figure 6.12 Combined AFM and Confocal Analysis

Superposition of AFM images with Actin edges (A), Tubulin edges (B) and holes (C). White contours indicate holes. See text for further explanation. Binarization of Confocal and AFM Images: Actin (green, D), tubulin (red, E), AFM (dark blue, F). G-I and J-L: Two other examples of Binarization. Scale bar: 5 µm.

6.4.4 Filopodia analysis

In order to analyze the properties of the filopodia that are seen present in the AFM scans, we characterized the length, mean diameter, base area and height of the filopodia. As can be seen in Figure 6.13, the base area of observed filopodia in either PFA or GLU fixed growth cones ranged from 0.1 to 3.22 µm^2 (Figure 6.13B). The distance between the base and the tip of a filopodium ranged from 0.5 to 8.6 µm (Fig. Figure 6.13C). The mean diameter of the filopodia varied from 113 to 887 nm (Figure 6.13D), while the mean height varied from 40 to 270 nm (Figure 6.13). The total length of the filopodia varied from 0.5 to 8.9 µm (Figure 6.13F). In Figure 6.13G, the height of a filopodium is plotted against its width, and there is no correlation between these two parameters (correlation coefficient=0). On the other hand, the plot of the area of a filopodium against its width (Figure 6.13H) shows a positive correlation with the value of the correlation coefficient being 0.5. Figure 6.13I shows the plot of the area against the height where there is a clustering at around the 100-200 nm range.

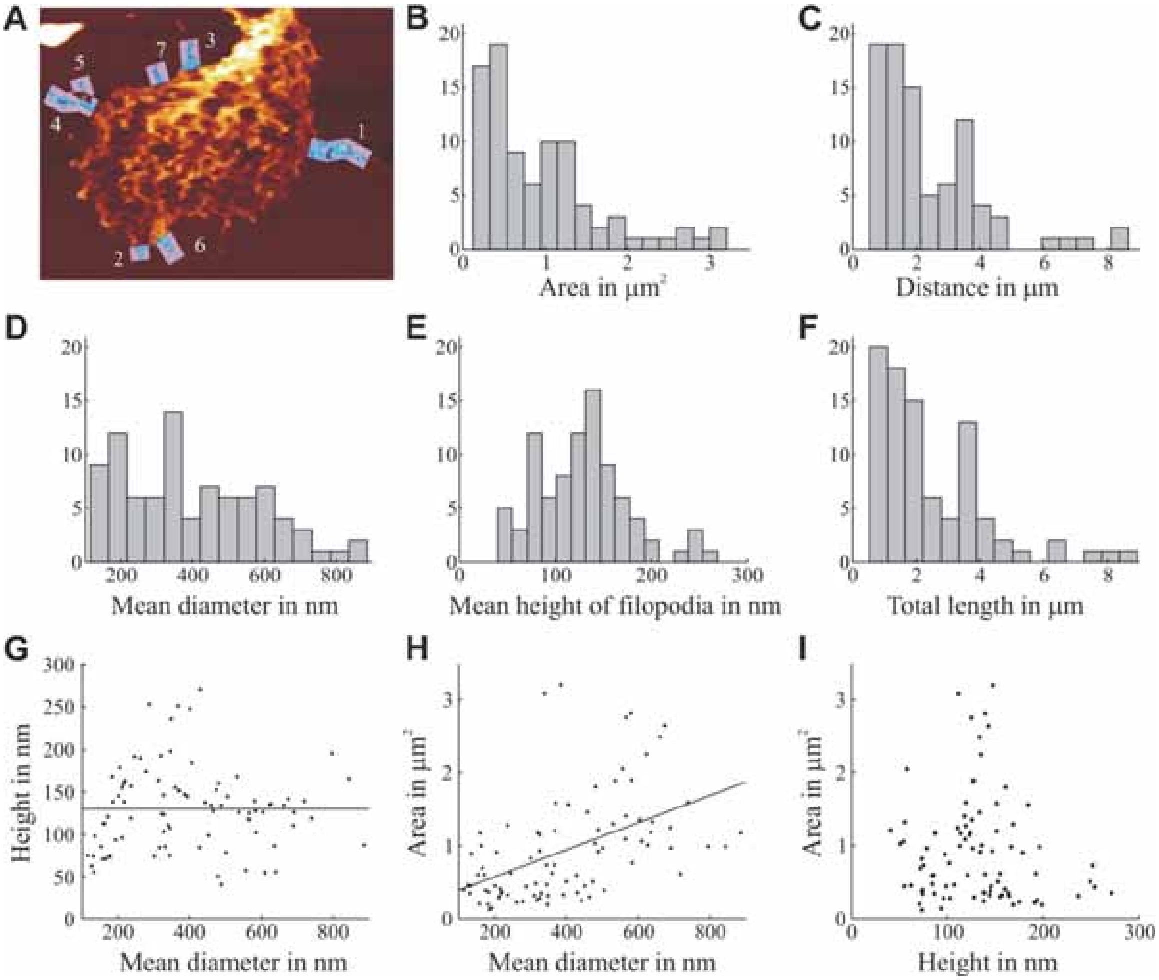


Figure 6.13 Filopodia analysis

A: AFM image with filopodia. B: Distribution of the area (in µm²) of the filopodia. C: Distribution of distance between the starting point and the end point (in µm). D: Distribution of the mean diameter (in µm) of the filopodia. E: Distribution of the mean height of the filopodia (in nm) from the AFM scan. F: Distribution of the total length of filopodia (in µm). G: Relation between the height (in nm) and the diameter (in nm) of the filopodia. H: Relation between the area (in µm²) and the diameter (in nm) of the filopodia. I: Relation between the area (in µm²) and the height (in nm) of the filopodia.

6.5 Discussion

Previous reports using AFM and confocal microscopy [61;62] have emphasized the need for an effective integration of the information provided by these techniques. In the present manuscript we provided a suitable method for solving this requirement. Taking advantage of tools used in Computer Vision we were able to superimpose images with different format and resolution. To our knowledge this is the first report where the images of the same cell obtained with different microscopes separately could be directly compared and superimposed. The confocal images with fluorescent staining of specific neuronal markers (such as neuronal class III β-tubulin, neural cell adhesion molecule (NCAM) or F-actin, a principal component of neuronal growth cones) and the AFM images of the 3-D structure (topography) of the same growth cone were obtained separately by the use of different instruments. The information obtained by the two techniques could be then fused together by the use of the method here described.

Point based registration is a suitable method for accurate matching of corresponding structures especially when the sample contains many corners and projections. However, further improvement can still be done to automatically detect corners, to make it easier for the user to select the candidates for correspondence. Matching with the contour is the selected method when there are no prominent corners. However it is not possible to use this method when the objects have steep walls or undercut regions. AFM scanning of these objects appear to have flared sides in the images (compare structures indicated by arrows in Figure 6.6A and C) due to the fact that AFM images are always a convolution of the original structure and the AFM tip shape.

Given the higher vertical resolution of AFM, novel structural components were discovered [62] while confocal microscopy was widely used in neuroscience (for a review see [87]. The possibility to combine the images from these two different analyses will increase the possibility to associate structure with the presence of sub-cellular components. Moreover the possibility to monitor cell behavior by time-laps confocal laser scanning microscopy will allow to link movement with structural changes (AFM) and the expression of specific sub-cellular components (confocal microscope).

The registration method here described is versatile and not limited to just AFM and CLSM images. Images taken from other modalities like CCD or SEM could be registered in a similar manner to get more understanding of the specimen under consideration. In future we plan to register the confocal layers taken at different depth with the 3D AFM topography to do in-depth analysis of the structure.

The present investigation of the shape and morphology of growth cones of differentiating DRG neurons reveals topographical structures with nanoscale dimensions which we refer as "invaginations" or "holes". These nanometric structures cannot be seen by conventional optical microscopes and are revealed by AFM.

The advantages of using Atomic Force Microscopy on biological samples have been extensively explored [58;59;62;88-90]. Information obtained from AFM provides an accurate description of the surface topography of the sample under investigation. We developed a procedure to reduce possible deformations of the surface caused by the AFM tip during scans [91]. This procedure has been referred to as the *0 pN AFM Image* of a sample. Using this method, it is possible to extrapolate an AFM image of a growth cone acquired at 0 pN scanning force as well as at any other desired force, if necessary, for comparison of AFM images acquired at different scanning forces.

Although there have been many reports on the use of AFM on biological samples, and on neurons in particular [57;58;89;90;92;93], we found novel structures with nanometric dimensions, which could not be seen with conventional optical microscopes and were not described in previous investigations with AFM. These novel nanometric structures are referred to as "invaginations" or "holes". These holes are reminiscent of those described by [72] and [73] in invivo retinal ganglion axons, presumably originating from the fusion or contact of lamellar extensions of the growth cone with glial processes. Holes observed in the present investigation are not caused by the AFM tip and are not introduced by the fixation procedure. Irrespective of the fixation procedure, these holes typically have a mean height ranging from 2 to 178 nm, with the base area ranging from 0.01 to 3.5 μm^2 with an average of 0.22 ± 0.20 μm^2 per growth Through polymerization, microtubules grow from the central region towards the peripheral region and where the plus ends interact with the filopodia coordinating cell movement. Comparison with LSCM images showed that more than 80% of the hole area lacks actin filaments or microtubules. Further, the mesh-like network of actin filaments and microtubules is confirmed through statistical analysis of the growth cones.

7 Conclusions and Perspectives

This thesis has described a complete framework for the problem of extracting shape from segmentation. The proposed approach builds upon ideas in image processing, computer vision and machine learning to provide a general, easy to use and fast method for Segmentation and texture classification. Our main contribution is the development of robust Link-method for region merging, and we have shown that our proposed method give state-of-the-art performance. The texture classification method uses a novel set of universal textons. These textons can be used for image segmentation and edge detection provided that the support is made small; the size of filters used currently is a bit large (13x13) for such applications.

Ongoing work from our laboratory (personal communication: Ban, J., Di Foggia, V., Migliorini, E., Lazzarino, M., Torre, V. and Ruaro, M. E.) on embryonic stem cells derived neurons in vitro suggests a correlation between the presence of the holes and the exploratory state of the growth cone. Growth cones of immobile isolated neurons or of retracting neurons display significant topological irregularities and holes. In contrast, actively exploring growth cones and those having established a contact with another growth cone rarely contain holes and have a smooth surface. Fragmentation seems a characteristic step during the physiological cycle of growth cones for example, when they stop actively exploring the environment. It is well known that cytoskeletal abnormalities are one of the hallmarks of neurodegenerative diseases [94;95] and it is possible that the kind of fragmentation described here for DRG growth cones also occurs during neurodegeneration or spinal cord injury [96]. Thus, it would be rather crucial to examine with AFM growth cones from neurons with such hallmarks of neurodegeneration.

7.1 Key Contributions

This thesis studies the question of obtaining a general and robust Image segmentation and a method to extract shapes for shape based object recognition. It further proposes a simple texture classification method. It also addresses the issues involved to register images taken from different modalities. It makes the following major contributions to these goals:

- The most important contribution is the introduction of scale edges. Normal gradient based edges, fails to detect texture edges and to compensate for this a very low threshold was needed to obtain a reasonable initial segmentation with many segments in the order of 1000. With the introduction of these edges, we were able to select high gradient and color edges, thus reducing the initial watershed segments to an order of 100-200.
- A method to reduce edges in the textured region is proposed, the idea is to use the information of fine texture edges only in one level of the multi-scale edge detection approach.
- A new merging procedure has been proposed, which is motivated from biology. Aggregation is the key to this approach. Small regions are merged first into bigger segments, and then many bigger regions are aggregated in one step, rather than merging them in pairs. This approach is especially important when the texture is not uniformly illuminated over the whole image.
- A method to eliminate high scale edges has been showed. This idea is also motivated from biology and photography. We do not perceive blurred edges (high scale edges). The significant segment boundaries pass along fine scale edges. Thus eliminating the false segment boundaries which are not passing through fine scale edges makes sense.

- An annotated database for mammals containing 6 classes is collected. Mammal classification especially cat and dog has been challenging. The database is challenging as the images contain object instances on a variety of background and have different poses. We wished to keep the task simple, motivated from the fact that introduction of simple database have led to discovery of novel algorithms in the past in the field of Segmentation (Berkeley dataset), texture recognition (Brodatz database), object recognition (Caltech21 & Caltech101) and shape recognition (Kimia dataset). A simple dataset allows easier challenge and once the results will achieve saturation, we may think of adding more difficult aspects.
- A texture classification method using only 3 filters is proposed, till date this is the minimum number of filters used for texture classification achieving top results.
- A new method for generating universal textons is implemented, which is generated only once and is database and texture-class independent. This also reduces the time during benchmarking as the traditional approach needed that the texton dictionary be built every time the training set was changed.
- A set of graphical user interfaces are implemented for image registration, for finding holes in the growth cones and for filopodia extraction.
- A method to generate a zero pN AFM image is shown to eliminate the dependence on the scanning force.

The thesis also provides guidelines on how to choose the various parameters of the feature sets and what effect each of these have on the overall classification performance. All of the conclusions are validated with extensive experimental tests.

7.2 Limitations

The proposed framework for segmentation has a number of limitations. A few of these are intrinsic shortcomings of the approach, while others relate to extensions that are worth investigating in the near future, and still others are open issues. This section presents the intrinsic limitations and the next section provides a discussion of the future work and open issues.

Although the proposed approach for segmentation gives good results for many images in natural scene, it is not suitable of applications like texture separation. The use of scale map shows us only the presence or absence of texture. Thus is not suitable for segmenting camouflage images. This can be seen from the segmentation results in Figure 7.1 Limitations of the segmentation methodFigure 7.1

Figure 7.1 Limitations of the segmentation method

Segmentation procedure has a relatively large number of parameters that need to be optimized and the parameter space is too large to allow every possible combination to be tested.

The most computationally intensive part of our framework for segmentation is edge detection,

The shape extraction algorithm assumes that there is a presence of only one object instance in the image; it was motivated from the fact that humans use an object centric approach to perform segmentation.

The texture feature used for texture classification is not robust to geometric variations in texture.

7.3 Future work

The segmentation algorithm can be improved by adding top-down information such as shape information. It will be also interesting to see if the current framework can be applied using graph-cut algorithms.

The shape extraction method is very naive, there is a plenty of scope for improvement. Clearly from the results, shape recognition is the key for mammal classification. Recently there has been a work to determine the salient object from the image background [97]. A set of novel features including multi-scale contrast, center-surround histogram, and color spatial distribution are used to describe a salient object locally, regionally, and globally. The work of them can be easily extended to determine saliency of segments instead of pixels. Figure 7.2 shows the saliency map implemented based on only color spatial distribution. The work is still preliminary and works if there is a considerable over segmentation. We are working to apply it when we reach to our final segmentation with say approximately 20 segments.

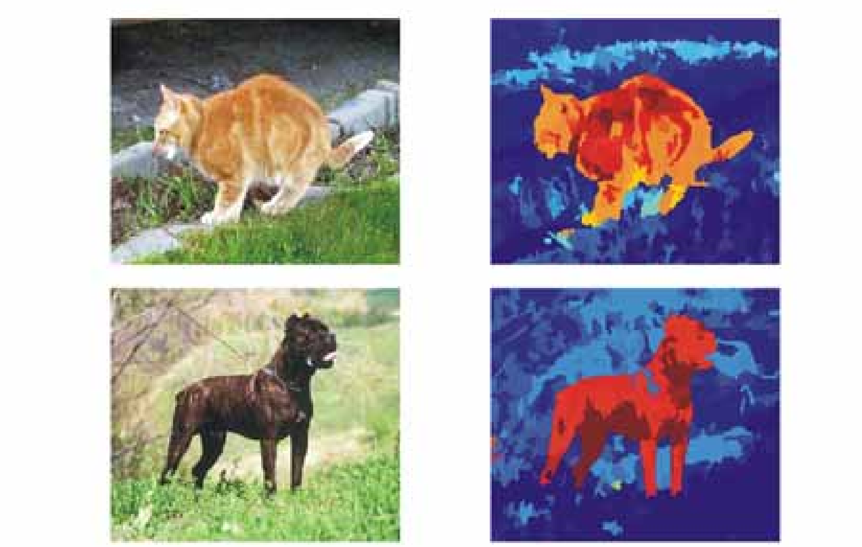


Figure 7.2 Saliency based on Color spatial distribution

The texture classification method should be made robust by adding some geometric invariant property, or should be used in combination with features like SIFT. With the use of our simple method combination of detectors is certainly feasible.

Also other filters can be tried to see if the texture recognition can improve but the real challenge is to make the texton-based method geometrically invariant.

For the growth cone study, it would be interesting to compare the hole statistics from a healthy model to that with a neurodegenerative one. Registration of entire confocal stack can help in mapping the exact texture map on the surface of the growth cone.

A Hardware & Software

For the experiments two computer types have been used:

- Intel Pentium IV, 3 GHz, 1 GB of RAM, running Windows XP & SUSE Linux 10.2
- Intel Core 2 Duo CPU, 1.5 GHz, using Windows Vista

The algorithms were executed / compiled with Matlab 7.0 R14 and Matlab 7.1 for Windows & Linux

CorelDraw 11 and ImageJ for creating figures.

Microsoft office Word 2003 for editing the thesis.

Reference Manager for citing the articles.

B Publications

T. Gritsun, S. Vogt, G. Engler, C. K. E. Moll, A. K. Engel, **S. Kondra**, L. Ramrath, and U. G. Hofmann. A simple microelectrode bundle for deep brain recordings. In *Neural Engineering, 2007. CNE '07. 3rd International IEEE/EMBS Conference on*, pages 114–117, 2007.

S. Kondra, W. Vanzella, R. Daliri, and V. Torre. A new algorithm for image segmentation by using color, intensity, and scale. *Perception* **36** ECVP Abstract Supplement, page 0. Pion Ltd, 2007.

R. Daliri, **S. Kondra**, W. Vanzella, and V. Torre. Cat - dog categorization in natural images. *Perception* **36** ECVP Abstract Supplement, page 0. Pion Ltd, 2007.

S. Kondra and V. Torre. Texture classification using three circular filters. In *Computer Vision, Graphics & Image Processing, 2008. ICVGIP '08. Sixth Indian Conference on*, pages 429–434, 2008.

S. Kondra, J. Laishram, J. Ban, E. Migliorini, V. Difoggia, M. Lazzarino, V. Torre, and M. Ruaro. Integration of confocal and atomic force microscopy images. *Journal of Neuroscience Methods*, 177(1):94–107, February 2009.

J. Laishram, **S. Kondra**, D. Avossa, E. Migliorini, M. Lazzarino and V. Torre A morphological analysis of growth cones of DRG neurons combining Atomic Force and Confocal Microscopy. (Under review), *submitted to Journal of structural biology,* submitted January 2009

List of Tables

List of Figures

Bibliography

[1] T. Leung and J. Malik, "Representing and Recognizing the Visual Appearance of Materials using Three-dimensional Textons," *Int. J. Comput. Vision*, vol. 43, no. 1, pp. 29-44, June2001.

[2] B. Julesz, "Textons, the elements of texture perception, and their interactions," *Nature*, vol. 290, no. 5802, pp. 91-97, Mar.1981.

[3] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics,", 2 ed 2001, pp. 416-423.

[4] S. Palmer, *Vision Science: Photons to Phenomenology* {The MIT Press}, 1999.

[5] Trucco and A. Verri, *Introductory Techniques for 3-D Computer Vision* Prentice Hall, 1998.

[6] M. Wertheimer, *Laws of organization in perceptual forms* 1938.

[7] R. C. Dubes and A. K. Jain, "Algorithms for Clustering Data," *Prentice Hall*, vol. 355, p. 356, 1988.

[8] M. Galun, E. Sharon, R. Basri, and A. Brandt, "Texture segmentation by multiscale aggregation of filter responses and shape elements," 2003, pp. 716-723.

[9] L. Lucchese and S. K. Mitra, "Colour image segmentation: a state-of-the-art survey," *PROCEEDINGS-INDIAN NATIONAL SCIENCE ACADEMY PART A*, vol. 67, no. 2, pp. 207-222, 2001.

[10] J. Shi and J. Malik, "Normalized cuts and image segmentation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 22, no. 8, pp. 888-905, 2000.

[11] C. Fowlkes, D. Martin, and J. Malik, "Learning Affinity Functions for Image Segmentation: Combining Patch-based and Gradient-based Approaches," *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, vol. 2 2003.

[12] N. Paragios and R. Deriche, "Geodesic Active Regions and Level Set Methods for Supervised Texture Segmentation," *International Journal of Computer Vision*, pp. 223-247, Feb.2002.

[13] W. Vanzella and V. Torre, "A versatile segmentation procedure," *IEEE transactions on systems, man, and cybernetics. Part B, Cybernetics : a publication of the IEEE Systems, Man, and Cybernetics Society*, vol. 36, no. 2, pp. 366-378, Apr.2006.

[14] B. Micusik and A. Hanbury, "Automatic Image Segmentation by Positioning a Seed," 2006, pp. 468-480.

[15] B. Micusik and T. Pajdla, "Multi-label image segmentation via max-sum solver," 2007, pp. 1-6.

[16] P. Felzenszwalb and D. Huttenlocher, "Image segmentation using local variation," 1998.

[17] E. Sharon, M. Galun, D. Sharon, R. Basri, and A. Brandt, "Hierarchy and adaptivity in segmenting visual scenes," *Nature*, vol. 442, no. 7104, pp. 810-813, June2006.

[18] A. Hanbury and B. Marcotegui, "Waterfall Segmentation of Complex Scenes," 2006, pp. 888-897.

[19] J. Winn and N. Jojic, "LOCUS: learning object classes with unsupervised segmentation,", 1 ed 2005, pp. 756-763.

[20] E. Borenstein, E. Sharon, and S. Ullman, "Combining Top-Down and Bottom-Up Segmentation," 2004, p. 46.

[21] J. Shotton, J. Winn, C. Rother, and A. Criminisi, "TextonBoost: Joint Appearance, Shape and Context Modeling for Multi-class Object Recognition and Segmentation," 2006, pp. 1-15.

[22] S. Wolfson, "Examining edge- and region-based texture analysis mechanisms," *Vision Research*, vol. 38, no. 3, pp. 439-446, Feb.1998.

[23] F. A. Pellegrino, W. Vanzella, and V. Torre, "Edge detection revisited," *Systems, Man, and Cybernetics, Part B, IEEE Transactions on*, vol. 34, no. 3, pp. 1500-1518, 2004.

[24] T. Lindeberg, "Edge Detection and Ridge Detection with Automatic Scale Selection," *International Journal of Computer Vision*, vol. 30, no. 2, pp. 117-156, Nov.1998.

[25] W. Vanzella, F. A. Pellegrino, and V. Torre, "Self-adaptive regularization," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 26, no. 6, pp. 804-809, 2004.

[26] P. Majer, "The Influence of the λ Parameter on Feature Detection with Automatic Scale Selection," 2001, pp. 245-254.

[27] D. R. Martin, C. C. Fowlkes, and J. Malik, "Learning to detect natural image boundaries using local brightness, color, and texture cues," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 26, no. 5, pp. 530-549, 2004.

[28] L. Vincent and P. Soille, "Watersheds in digital spaces: an efficient algorithm based on immersion simulations," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 13, no. 6, pp. 583-598, 1991.

[29] M. Sonka, V. Hlavac, and R. Boyle, *Image Processing: Analysis and Machine Vision* Thomson-Engineering, 1998.

[30] E. Borenstein and S. Ullman, "Class-Specific, Top-Down Segmentation," 2002, pp. 639-641.

[31] A. Criminisi, "Microsoft research cambridge object recognition image database http://research.microsoft.com/vision/cambridge/recognition/," 2004.

[32] M. Fink and S. Ullman, "From Aardvark to Zorro: A Benchmark for Mammal Image Classification," *International Journal of Computer Vision*, vol. 77, no. 1, pp. 143-156, May2008.

[33] M. Daliri and V. Torre, "Robust symbolic representation for shape recognition and retrieval," *Pattern Recogn.*, vol. 41, no. 5, pp. 1799-1815, 2008.

[34] D. Nister and H. Stewenius, "Scalable Recognition with a Vocabulary Tree,", 2 ed 2006, pp. 2161-2168.

[35] E. Nowak, F. Jurie, and B. Triggs, "Sampling Strategies for Bag-of-Features Image Classification," in *Computer Vision, ECCV* 2006, pp. 490-503.

[36] D. Forsyth and J. Ponce, *Computer Vision: A Modern Approach* {Prentice Hall}, 2002.

[37] J. De Bonet, "Multiresolution Sampling Procedure for Analysis and Synthesis of Texture Images," *Computer Graphics*, vol. 31, no. Annual Conference Series, pp. 361-368, 1997.

[38] O. Cula and K. Dana, "Compact Representation of Bidirectional Texture Functions," *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, vol. 1 2001.

[39] S. Konishi and A. L. Yuille, "Statistical cues for Domain Specific Image Segmentation with Performance Analysis," *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, vol. 1 2000.

[40] J. Portilla and E. Simoncelli, "A Parametric Texture Model Based on Joint Statistics of Complex Wavelet Coefficients," *Int. J. Comput. Vision*, vol. 40, no. 1, pp. 49-70, Oct.2000.

[41] M. Varma and A. Zisserman, "Classifying Images of Materials: Achieving Viewpoint and Illumination Independence," 2002, pp. 255-271.

[42] T. Rikert, M. Jones, and P. Viola, "A Cluster-based Statistical Model for Object Detection," 1999, pp. 1046-1053.

[43] D. Heeger and J. Bergen, "Pyramid-based texture analysis/synthesis," ACM, 1995, pp. 229-238.

[44] M. Varma and A. Zisserman, "Texture classification: are filter banks necessary?,", 2 ed 2003, pp. 691-698.

[45] K. Dana, B. van Ginneken, S. Nayar, and J. Koenderink, "Reflectance and texture of real-world surfaces," *ACM Trans. Graph.*, vol. 18, no. 1, pp. 1-34, Jan.1999.

[46] E. Hayman, B. Caputo, M. Fritz, and J. O. Eklundh, "On the Significance of Real-World Conditions for Material Classification," 2004, pp. 253-266.

[47] S. Lazebnik, C. Schmid, and J. Ponce, "A sparse texture representation using local affine regions," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 27, no. 8, pp. 1265-1278, 2005.

[48] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid, "Local Features and Kernels for Classification of Texture and Object Categories: A Comprehensive Study," 2006, p. 13.

[49] D. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91-110, Nov.2004.

[50] B. Schoelkopf and A. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond (Adaptive Computation and Machine Learning)* {The MIT Press}, 2001.

[51] K. Grauman and T. Darrell, "The pyramid match kernel: discriminative classification with sets of image features,", 2 ed 2005, pp. 1458-1465.

[52] P. Brodatz, *Textures: A Photographic Album for Artists and Designers* Dover Publications, 1999.

[53] C. Schmid, "Constructing models for content-based image retrieval,", 2 ed 2001, p. II-45.

[54] R. Duda and P. Hart, *Pattern Classification and Scene Analysis* {John Wiley & Sons Inc}.

[55] C. C. Chang and C. J. Lin, "LIBSVM: a Library for Support Vector Machines http://www.csie.ntu.edu.tw/~cjlin/libsvm," 2001.

[56] B. S. Manjunath and W. Y. Ma, "Texture features for browsing and retrieval of image data," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 18, no. 8, pp. 837-842, 1996.

[57] V. Parpura, P. G. Haydon, and E. Henderson, "Three-dimensional imaging of living neurons and glia with the atomic force microscope," *Journal of cell science*, vol. 104 (Pt 2), pp. 427-432, Feb.1993.

[58] A. Bonfiglio, "Subcellular Details of Early Events of Differentiation Induced by Retinoic Acid in Human Neuroblastoma Cells Detected by Atomic Force Microscope," *Experimental Cell Research*, vol. 216, no. 1, pp. 73-79, Jan.1995.

[59] C. Reddy, K. Malinowska, N. Menhart, and R. Wang, "Identification of TrkA on living PC12 cells by atomic force microscopy," *Biochimica et Biophysica Acta (BBA) - Biomembranes*, vol. 1667, no. 1, pp. 15-25, Nov.2004.

[60] G. Li, N. Xi, and D. Wang, "Probing membrane proteins using atomic force microscopy," *Journal of Cellular Biochemistry*, vol. 97, no. 6, pp. 1191-1197, 2006.

[61] B. Rajwa, H. A. McNally, P. Varadharajan, J. Sturgis, and J. P. Robinson, "AFM/CLSM data visualization and comparison using an open-source toolkit," *Microscopy research and technique*, vol. 64, no. 2, pp. 176-184, June2004.

[62] H. Mcnally, B. Rajwa, J. Sturgis, and J. Robinson, "Comparative three-dimensional imaging of living neurons with confocal and atomic force microscopy," *Journal of Neuroscience Methods*, vol. 142, no. 2, pp. 177-184, Mar.2005.

[63] D. Hill, P. Batchelor, M. Holden, and D. Hawkes, "Medical image registration," *Physics in Medicine and Biology*, vol. 46, no. 3, p. R1-R45, 2001.

[64] H. Lester and S. Arridge, "A survey of hierarchical non-linear medical image registration," *Pattern Recognition*, vol. 32, no. 1, pp. 129-149, Jan.1999.

[65] A. Maintz and M. Viergever, "A survey of medical image registration," *Medical Image Analysis*, vol. 2, no. 1, pp. 1-36, Mar.1998.

[66] P. A. van den Elsen, E. J. D. Pol, and M. A. Viergever, "Medical image matching-a review with classification," *Engineering in Medicine and Biology Magazine, IEEE*, vol. 12, no. 1, pp. 26-39, 1993.

[67] E. Kandel, J. Schwartz, and T. Jessell, *Principles of Neural Science* {McGraw-Hill Medical}, 2000.

[68] J. Howard, *Mechanics of Motor Proteins and the Cytoskeleton* {Sinauer Associates}, 2001.

[69] U. Hofmann, "Investigating the Cytoskeleton of Chicken Cardiocytes with the Atomic Force Microscope," *Journal of Structural Biology*, vol. 119, no. 2, pp. 84-91, July1997.

[70] T. Kuznetsova, M. Starodubtseva, N. Yegorenkov, S. Chizhik, and R. Zhdanov, "Atomic force microscopy probing of cell elasticity," *Micron*, vol. 38, no. 8, pp. 824-833, Dec.2007.

[71] G. Kirfel, A. Rigort, B. Borm, and V. Herzog, "Cell migration: mechanisms of rear detachment and the formation of migration tracks," *European Journal of Cell Biology*, vol. 83, no. 11-12, pp. 717-724, Dec.2004.

[72] P. Godement, L. C. Wang, and C. A. Mason, "Retinal axon divergence in the optic chiasm: dynamics of growth cone behavior at the midline," *The Journal of neuroscience : the official journal of the Society for Neuroscience*, vol. 14, no. 11 Pt 2, pp. 7024-7039, Nov.1994.

[73] C. Mason and L. Erskine, "Growth cone form, behavior, and interactions in vivo: Retinal axon pathfinding as a model," *Journal of Neurobiology*, vol. 44, no. 2, pp. 260-270, 2000.

[74] G. Binnig, C. F. Quate, and C. Gerber, "Atomic Force Microscope," *Physical Review Letters*, vol. 56, no. 9, p. 930, Mar.1986.

[75] R. Lal and S. A. John, "Biological applications of atomic force microscopy," *Am J Physiol Cell Physiol*, vol. 266, no. 1, pp. C1-21, Jan.1994.

[76] D. J. Mueller, D. Fotiadis, S. Scheuring, S. A. Mueller, and A. Engel, "Electrostatically balanced subnanometer imaging of biological specimens by atomic force microscope," *Biophysical journal*, vol. 76, no. 2, pp. 1101-1111, Feb.1999.

[77] S. Scheuring, J. Seguin, S. Marco, D. L+¬vy, B. Robert, and J. L. Rigaud, "Nanodissection and high-resolution imaging of the Rhodopseudomonas viridis photosynthetic core complex in native membranes by AFM. Atomic force microscopy," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 100, no. 4, pp. 1690-1693, Feb.2003.

[78] D. Cojoc, F. Difato, E. Ferrari, R. B. Shahapure, J. Laishram, M. Righi, E. M. Di Fabrizio, and V. Torre, "Properties of the force exerted by filopodia and lamellipodia and the involvement of cytoskeletal components," *PLoS ONE*, vol. 2, no. 10 2007.

[79] M. Dailey, G. Marrs, J. Satz, and M. Waite, "Concepts in imaging and microscopy. Exploring biological structure and function with confocal microscopy," *The Biological bulletin*, vol. 197, no. 2, pp. 115-122, Oct.1999.

[80] B. Zitova and J. Flusser, "Image registration methods: a survey," *Image and Vision Computing*, vol. 21, no. 11, pp. 977-1000, Oct.2003.

[81] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham, "Active shape models - their training and application," *Comput. Vis. Image Underst.*, vol. 61, no. 1, pp. 38-59, Jan.1995.

[82] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *International Journal of Computer Vision*, vol. V1, no. 4, pp. 321-331, Jan.1988.

[83] C. Xu and J. L. Prince, "Snakes, shapes, and gradient vector flow," *Image Processing, IEEE Transactions on*, vol. 7, no. 3, pp. 359-369, 1998.

[84] B. Barber, D. Dobkin, and H. Huhdanpaa, "The Quickhull Algorithm for Convex Hulls," *ACM Transactions on Mathematical Software*, vol. 22, no. 4, pp. 469-483, 1996.

[85] A. R. Cross and R. C. Williams, "Kinky microtubules: bending and breaking induced by fixation in vitro with glutaraldehyde and formaldehyde," *Cell motility and the cytoskeleton*, vol. 20, no. 4, pp. 272-278, 1991.

[86] S. Kondra, J. Laishram, J. Ban, E. Migliorini, V. Difoggia, M. Lazzarino, V. Torre, and M. Ruaro, "Integration of confocal and atomic force microscopy images," *Journal of Neuroscience Methods*, vol. 177, no. 1, pp. 94-107, Feb.2009.

[87] C. M. Niell and S. J. Smith, "Live optical imaging of nervous system development," *Annual review of physiology*, vol. 66, pp. 771-798, 2004.

[88] D. Fotiadis, S. Scheuring, S. Müller, A. Engel, and D. Müller, "Imaging and manipulation of biological structures with the AFM," *Micron (Oxford, England : 1993)*, vol. 33, no. 4, pp. 385-397, 2002.

[89] S. Yunxu, L. Danying, R. Yanfang, H. Dong, and M. Wanyun, "Three-dimensional structural changes in living hippocampal neurons imaged using magnetic AC mode atomic force microscopy," *Journal of electron microscopy*, vol. 55, no. 3, pp. 165-172, June2006.

[90] E. Grzywa, A. Lee, G. Lee, and D. Suter, "High-resolution analysis of neuronal growth cone morphology by comparative atomic force and optical microscopy," *Journal of Neurobiology*, vol. 66, no. 14, pp. 1529-1543, 2006.

[91] S. Sen, S. Subramanian, and D. E. Discher, "Indentation and adhesive probing of a cell membrane with AFM: theoretical model and experiments," *Biophysical journal*, vol. 89, no. 5, pp. 3203-3213, Nov.2005.

[92] H. Ohshiro, R. Suzuki, T. Furuno, and M. Nakanishi, "Atomic force microscopy to study direct neurite-mast cell (RBL) communication in vitro," *Immunology letters*, vol. 74, no. 3, pp. 211-214, Nov.2000.

[93] R. Z. Zhang, W. Y. Zhu, M. Y. Xia, and Y. Feng, "Morphology of cultured human epidermal melanocytes observed by atomic force microscopy," *Pigment cell research / sponsored by the European Society for Pigment Cell Research and the International Pigment Cell Society*, vol. 17, no. 1, pp. 62-65, Feb.2004.

[94] J. E. Goldman and S. H. Yen, "Cytoskeletal protein abnormalities in neurodegenerative diseases," *Annals of neurology*, vol. 19, no. 3, pp. 209-223, Mar.1986.

[95] C. T. McMurray, "Neurodegeneration: diseases of the cytoskeleton?," *Cell death and differentiation*, vol. 7, no. 10, pp. 861-865, Oct.2000.

[96] V. Tom, M. Steinmetz, J. Miller, C. Doller, and J. Silver, "Studies on the Development and Behavior of the Dystrophic Growth Cone, the Hallmark of Regeneration Failure, in an In Vitro Model of the Glial Scar and after Spinal Cord Injury," *J. Neurosci.*, vol. 24, no. 29, pp. 6531-6539, July2004.

[97] T. Liu, J. Sun, N. N. Zheng, X. Tang, and H. Y. Shum, "Learning to Detect A Salient Object," 2007, pp. 1-8.