Declaration

I hereby declared that I'm the sole author of this thesis and this thesis is my own original work. It contains no material that has been accepted for the award of any other degree or diploma in any university or other institution.

Loke Kar Seng

Notice 2

I certify that I have made all reasonable efforts to secure copyright permissions for third-party content included in this thesis and have not knowingly added copyright content to my work without the owner's permission.

Image Segmentation by Object Contour Extraction: A top down and bottom up approach combining object recognition and local cues.

By

Loke Kar Seng

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

September 2011

This thesis is dedicated

to

Holly

Acknowledgement

I would like to thank Simon Egerton, David Squire and Lee Poh Aun for their support, advice and encouragement. And thanks to Rebecca Wong for her eagle eyes.

Publication List

Journals

1. K.W. See, K.S. **Loke**, P.A. Lee and K.F. Loe (2007), "Image Reconstruction using Various Discrete Orthogonal Polynomials in Comparison with DCT", Journal of Appl. Mathematics and Computation. Volume 193, Issue 2, 1 November 2007. pp. 346-359
2. **Loke** K.S. & Mylini Munusamy (2010). Preliminary Survey on Visual/Verbal Dimension In Feldman-Soloman Model and Assessment Format in an Introductory Programming Course. Computers in Education Journal 1(2). April-June 2010. pp. 52-61

Conferences

1. **Loke** K.S. (2003). Hybrid Stroke/Vowel Input System for Mobile Devices. OZCHI 2003. Brisbane Australia,26th-28th November.
2. **Loke**, K.S (2004). A Brief Survey of Description Logic for Maps and Diagrams. 1st International Conference on Informatics 2004. Kuala Lumpur.
3. **Loke** K.S (2005). “Notes on Defining Novelty for Computational Creativity”. IJCAI 2005 Edinburgh. Computational Creativity Workshop. 2005
4. **Loke** Kar Seng and Chan Pei Pei (2006). “Online News Classification for Individual Stock Movement Predictions: A Malaysian Market Study”. SME-Entrepreneurship Global Conference 2006. Malaysia. Monash University Malaysia.
5. **Loke** Kar Seng & Tan Choon Ling (2006). “Diagram Matching based on the Dual Ant Colony Optimization Algorithm”. In Advances in Natural Computation and Data Mining. ICNC 2006. Xian University Press.
6. K.W. See, K.S. **Loke**, P.A. Lee and K.F. Loe (2007), "Comparative Study of Image Reconstruction using Discrete Orthogonal Polynomials", Conference on IT Research and Applications 2007 (CITRA 2007), 4-5 April, 2007, Kuala Lumpur, Malaysia.
7. Teo Y.J. & **Loke** K.S. (2007), "Intelligent Load Balancing Algorithm for a Mobile Agent System", Conference on IT Research and Applications (CITRA 2007), 4-5 April, Selangor, Malaysia.
8. Marc Cheong & Kar-Seng **Loke**.(2008). “Textile Recognition Using Tchebichef Moments of Co-occurrence Matrices”. ICIC 2008. LNCS 5226. pp. 1044-1051.
9. Marc Cheong & Kar-Seng **Loke**.(2008). “An Approach to Texture-Based Image Recognition by Deconstructing Multispectral Co-occurrence Matrices using Tchebichef Orthogonal Polynomials”. ICPR 2008, December 8-11 Tampa, Florida.
10. **Loke** K.S. & Khong W.S. (2009). “Real-time Video Hand Gestures Recognition in Cluttered Background”. Malaysia Joint Conference on Artificial Intelligence MJCAI '09. 14-16 July 2009 Kuala Lumpur. ISBN: 978-983-42887-2-3.
11. **Loke** K.S. & Marc Cheong (2009). “Efficient Textile Recognition via Decomposition of Co-occurrence Matrices.” IEEE International Conference on Signal & Image Processing ICSIPA '09. 18-19 Nov 2009. Kuala Lumpur.
12. **Loke** K.S. (2010) “Wedgelets-based object contour detection using ACO.” The 6th International Conference on Natural Computation & the 7th International Conference on Fuzzy Systems and Knowledge Discovery (ICNC'10-FSKD'10) 10-12 August 2010. Yantai, China.
13. **Loke** K.S. & Simon Egerton (2010). Scene Understanding: A Framework for Image Segmentation Via Object Recognition. Sixth International Conference on Intelligent Environments 2010. 19-21 July 2010. Kuala Lumpur Malaysia.

14. **Loke** K.S. & Simon Egerton (2010). Automated Eye on Nature (AEON) and the Were-Tigers of Belum. 1st International Workshop on Creative Science - Science Fiction Prototyping for Research Innovation. Sixth International Conference on Intelligent Environments 2010. 19-21 July 2010. Kuala Lumpur Malaysia.
15. **Loke** K.S., Egerton S., Cristofaro, D., & Clementson, S (2011). "Automated real-time dynamic identification of flying and resting butterfly species in the natural environment", International Conference on Environment Science and Engineering - ICESE 2011, Bali, April 2011.

Book Chapters

1. **Loke** K.S. (2009). An Approach to Textile Recognition. In Lazinica, A. (ed). Pattern Recognition. ISBN 978-953-7619-X-X. In-TECH Publications. pp. 339-360.

Abstract

Computer image understanding of pictorial information has many useful applications and is one of most researched areas in computer science. The process of automated image understanding often requires that the image is first automatically segmented. Image segmentation divides an image into coherent parts and has been predominantly performed by grouping regions of similarity or by partitioning the image based on edge detection. This form of segmentation is frequently unsatisfactory because it fails to segment at meaningful object boundaries. Therefore, our work is aimed at segmenting objects by extracting (or detecting) their boundaries. The segmentation of object boundaries and their identification can be considered as the first step towards semantic image scene understanding.

In this work we introduce a new image representation that combines wedgelets and discrete orthogonal polynomials to model straight edges and textures. We use an Ant Colony Optimization (ACO) algorithm to search for object contours that satisfy local constraints using local cues. The ACO algorithm performs a biased exploration in alternation with optimization allowing us to escape local maxima. However, the results show that local cues do not provide sufficient information to the Ant algorithm to enable it to construct viable object contours.

To overcome the local cue limitations, we examine the use of global cues, specifically the shape of the object being segmented, but this leads us to a potential paradox. If the purpose of segmentation is to obtain the identity of the object, then how can we obtain the shape of the object without first knowing its identity? We resolve this apparent paradox by using other methods to first identify the object and then use the result of the identification to guide the segmentation. Based on this concept we have developed a segmentation algorithm which first identifies the object using image patches and then segments the object using the active contour algorithm. The algorithm is rotation, translation and scale invariant.

However, using image-based cues for recognition has limitations for general categorization, as many objects are usually categorized by their shape. We solve this by developing a shape-based cue for object recognition that is biologically inspired. Shape-based cues allow us to recognize objects based on their global shape. Our experimental results show that our method is comparable or better than known methods. We then use the edge-based cues to construct the objects contour detection using the ACO algorithm developed earlier. In this sense our work combines a top-down, bottom-up segmentation approach that has parallels in human visual systems.

Contents

Chapter One: **Introduction**

Detecting objects from two-dimensional natural scene images remains a central problem in computer vision. The problem is compounded by issues of lighting conditions, shadows, occlusion, pose, view angle, including accidental matching of object's surface properties to the background. This research seeks to learn object contours from two-dimensional images and use them for image retrieval and scene understanding.

A basic computational approach is to look for the image edges as discontinuity in intensity. However, an object contour is more than just an intensity differential. Object contours are commonly extracted with one of the line edge detector image processing algorithms such as the Canny operator (Canny 1986). The response from such operators is rarely the object's outline contour. This is because an edge in the contour is not only a differential in intensity, but in general any form of discontinuity between the object and the background. This problem is complicated by camouflage and occlusion. The contour could reflect discontinuities in color and textures as well, and may take into consideration perceptual gestalt properties, prior knowledge and expectation. Contour detection is a global concept related to the meaning and recognition of *object from ground*[1]. The challenge is to integrate all these considerations into a viable model.

A computational approach may be augmented by ideas on how natural systems perform similar tasks. Natural human vision systems typically enlist multiple cues to combat ambiguity and to maximize reliability of boundary detection under a wide variety of circumstances (Zhou and Mel 2008).There is evidence that neurological features in the brain are attuned to shapes or contour information. The inferior temporal neurons have been suggested to be involved in the analysis of global boundary information, using local

[1] Separating the main foreground object from the background

edge information from the striate and prestriate[2] cortex to create the boundary representation (Schwartz, Desimone et al. 1983). Neurons in the visual cortex region V2 and V4 have also been known to respond to a variety of local contours. V2 units respond to angles, arcs and intersections (Hegde and Van Essen 2000; Hegde and Van Essen 2003), and V4 units are selective for a particular local shape configuration (e.g. sharp concavity) and the population coding can provide a description of the shape contour.

1.1 Research questions

Edges and contours are sometimes used interchangeably. For our purposes, we take edges to be lines derived primarily from low level edge detectors and are usually short. Edges linked together to obtain longer curvilinear lines are what we call contours, and they usually delineate some regions of similar features. In an image, naturally there will be a variety of edges and contours. Contours that have been identified to belong to a particular object, we call object contours. The aim of this research is to identify the object contours within an image belonging to a natural scene. Segmentation involves partitioning of an image into parts of similarity given some measure of similarity. The similarity measure could be based on intensity, colour, texture, or some other measure. After the segmentation, from the borders of these partitioned regions, the segmentation contours can be extracted. However, these contours may or may not encapsulate the entire object. Our objective is to just extract the object contour. We make one assumption that the object contour we extract is the primary, or foreground object. In this case, this may also be called foreground-background segmentation, if the actual contour is not being retrieved.

We wish to be able to extract contour information from a two-dimensional scene image automatically by integrating local and global information sufficiently for image retrieval by object contour. This can be seen as a first step to semantic image retrieval without depending on manual external textual annotation. Automated annotation is widely

[2] Chapter 4 provides an overview of the brain structure

recognized as a extremely difficult issue (Datta, Joshi et al. 2008). The difficulty stems from the problem of image understanding in the presence of complex backgrounds. The approach of classification of whole images via global visual signatures has been criticized as too coarse, as well as the use of laboratory condition images with simple backgrounds that can easily be segmented (Lew, Sebe et al. 2006). The current challenge, as stated in (Lew, Sebe et al. 2006), is to "detect all of the semantic content within an image such as faces, trees, animals, etc. with emphasis on the presence of complex backgrounds".

Identifying object contours can be a chicken and egg problem, since we need to know the object before we can obtain the contours and the contours are used to define the object. Current image contour extraction approaches do not identify the object to extract its contour because the objective is to use those contours to identify the object.

Our main research question centres on precisely how to detect and extract object contours from a natural scene image:

R1. How can a model of automatic object contour detection and extraction from scene images that includes contextual and possibly prior information improve significantly an object contour-based image retrieval system?

We will be considering natural scene images with a single object in a natural background. The use of natural scene image will avoid criticism leveled at specially constructed artificial environments that do have real world variations. We are working with single objects for computational purposes and to focus on the objective at hand.

Subsidiary questions:

R1.1 What type of contextual information or prior information that is appropriate?
R1.2 What is the interaction between object detection and contour extraction given that both are inter-dependent?

R1.3 What is one possible appropriate representation and properties of the object contour?

1.2 Overview of Image Segmentation, Contour Detection and Shape representation

Image segmentation is often a required process in most computer vision applications; it usually acts as a preliminary process to higher level tasks such as semantic object recognition or scene interpretation. These images are usually segmented by internal properties such as texture, colour or edges.

We sketch a broad overview of various major approaches to image segmentation. Image segmentation can be grouped into three general approaches, namely, edge-based, contour-based and region-based (Figure 1). Edge-based methods are generally based on edge detection and edge linking based on psychological and gestalt properties. Contour-based methods use the Active Contour paradigm; these will be discussed further in Chapter 3. Region-based approaches include approaches using clustering to graph, diffusion and variational methods. Many of these approaches do not account for object boundaries in the segmentation, taking for granted that the low-level[3] properties somehow coincide with object boundaries. It is only in recent years that object-centered segmentation has come to the fore.

A common approach is to use intensity gradients to obtain an intensity-based edge (Marr and Hildreth 1980; Hildreth 1983; Canny 1986; Ziou and Tabbone 1998), which is then used to estimate actual object edges (Frei and Chen 1977; Martens 1997) in the image. Color (Ruzon and Tomasi 2001; Weijer, Vevers et al. 2005), energy maxima of quadrature filter pairs (Folsom and Pinter 1998), or phase energy (Morrone and Burr 1988) can also be used to locate the edges. Another approach is to analyze the local region for directional orientation using different operators (Martens 1997; Jiang 2007).

[3] Essentially pixel-level properties as opposed to global properties of the object

Texture information is also useful (Chaji and Ghassemian 2006; Hidayat and Green 2009) in helping to determine edges.

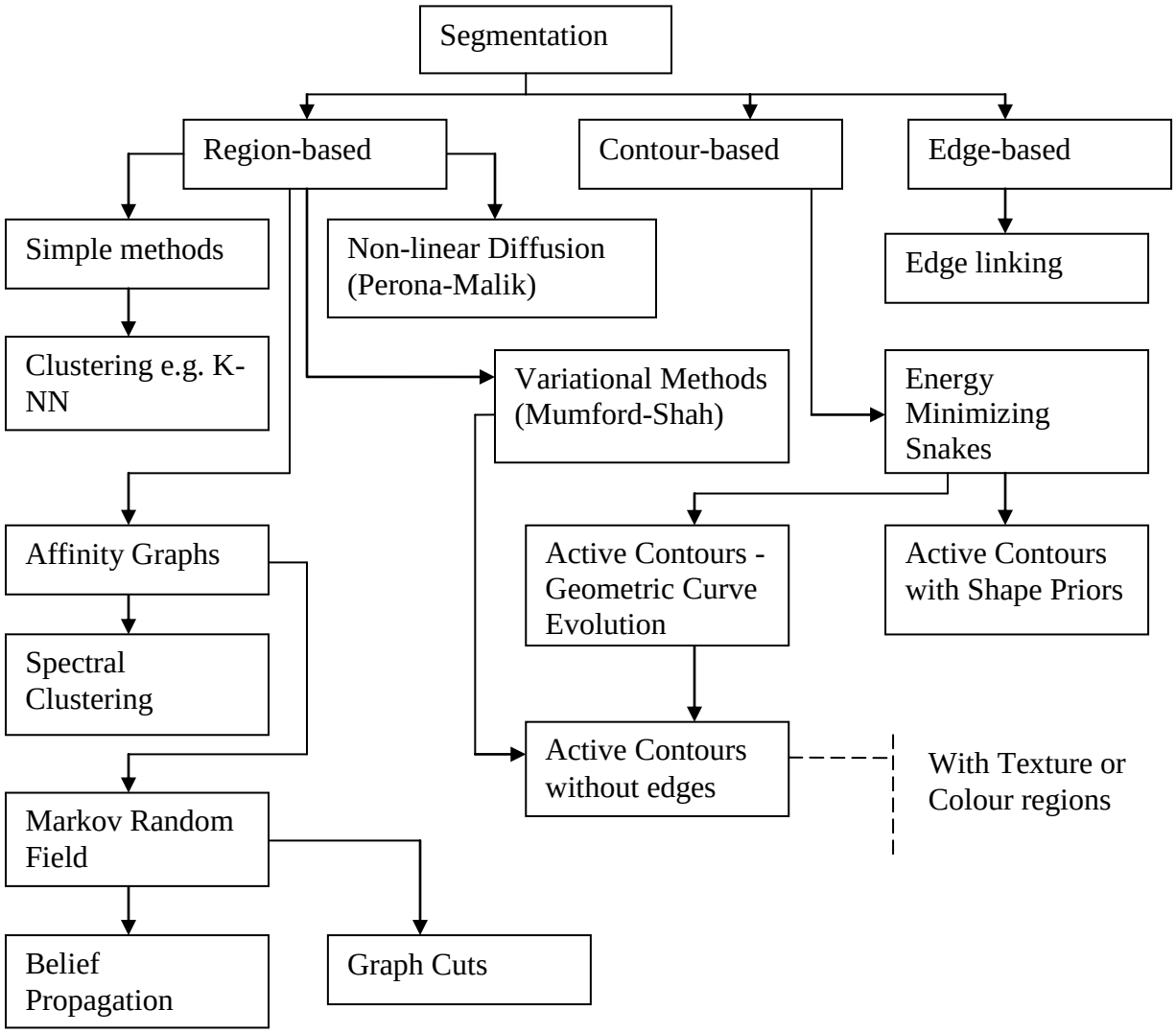


Figure 1. Approaches to Image Segmentation

Another approach takes inspiration from biology, mimicking the center-surround receptive field neurons of the human visual system (Joshi and Sivaswamy 2006; Papari, Campisi et al. 2006; Papari, Campisi et al. 2007; Papari, Campisi et al. 2007). The approach utilizes intensity gradient computations to obtain the initial edge map. It is post

processed by an excitatory center and inhibitory surround mechanism around the pixel edge point. The surround inhibition suppresses texture and spurious isolated edges by comparing local information around the edge point.

These edge detectors provide a set of primitives that can be used to form more elaborate models using chains, lines, circles and splines. The gradient direction from edge detectors could also be used to fit a set of cubic-splines to obtain the object's edges (Chen and Yang 1995). A review of various gradient intensity methods (Heath, Sarkar et al. 1996) found that the popular Canny operator (Canny 1986) does not perform well against human judgment. It is often unreliable at determining contours especially within an even moderately complex background.

Edges from gradient information are prone to spurious edges and tiny branching lines, caused by textured regions, shadows and illumination; hence they are unreliable in determining contours or boundaries of objects. Several methods have been proposed to suppress these spurious edges, either by suitable preprocessing, post processing or considering additional local or contextual information.

Anisotropic diffusion (Perona and Malik 1990) is an example of an approach that utilizes preprocessing to improve edge detection by enhancing edges and smoothing within estimated edge boundaries, after which a Canny edge detector can be used. Multi-scale methods (Williams and Shah 1990; Ziou and Tabbone 1993) have also been used to determine relevant edges. Other works seek to obtain a confidence measure of the edge by comparing it to an ideal edge (Meer and Georgescu 2001), or by (Ando 2000) using a covariance measure of the image intensity to determine edges and corners.

Usually the prerequisite for finding object contours begins by finding edges and suitably post processing them. Many of these techniques use the gradient-based or energy-based edge detector, supplemented by pre or post processing techniques to clean up spurious edges that are not part of the object contour. The issue is determining which of these

edges are legitimate, and this itself is not a simple problem. Even with legitimate edges remaining after preprocessing, there remains the task ahead of assigning them to object contours. However, edges, as obtained via intensity gradient differences, are quite different from object contours.

Saliency detection is used for determining important areas in the image. These are regions of interest that attracts our visual attention. Saliency detected images will allow us to focus on the object of interest and to discard corresponding edges that do not belong to the salient object. For example, in the image below (Figure 2), the edges from the background could be discarded if the bear is detected as a salient object.

Saliency-based or bottom up visual attention allow humans to detect in real time non-specific conspicuous targets from cluttered visual environments. This can be modeled by saliency maps that are composed of various low-level features such as intensity, color distribution, texture, multi-scale contrast, and center surround histogram (Itti and Koch 1999; Joshi and Sivaswamy 2005; Walther, Rutishauser et al. 2005; Moosmann, Larlus et al. 2006; Walther and Koch 2006; Liu, Sun et al. 2007). These features are fed to a learning procedure, e.g. using a biologically-inspired inhibition method (Itti and Koch 1999) or to create a conditional random field model (Liu, Sun et al. 2007). An alternate approach is to model perceptual “pop out” via attraction and repulsion measures (Yu and Shi 2001). It is know that only certain elementary visual features which are extracted early in the visual processing stream will “pop out” (Ramachandran and Rogers-Ramachandran 2008). It is likely that these effects have to be modeled if the aim is to make computer-based retrieval system comparable to the human visual system.

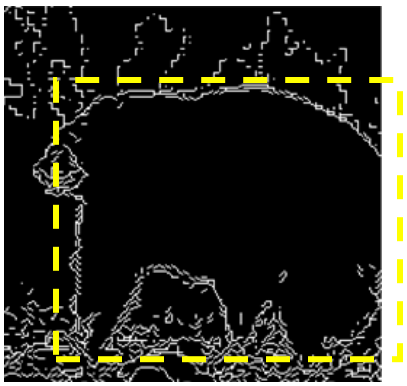


Figure 2: Saliency detection (example)

However, saliency detection will only reduce the number of edges to consider and cannot improve on the contour detection.

From the edges detected, they could be joined together to produce longer edge segments via some sort of edge-linking. Edge-linking approaches assume that it is possible to group a set of edges using exclusively geometric relationships and perceptual properties. The geometric cues that are frequently used are often based on Gestalt laws (Kanizsa 1979) using cues such as proximity, smoothness, continuity and compactness. Such cues are thought to be good predictors for edge linking because elements that exhibit such properties are unlikely to be accidental.

Ullman and Sha'ashua (Shashua and Ullman 1988) developed an edge-linking approach using what they called 'edge saliency'. They distinguished two types of edge saliency: local saliency which are local properties such as intensity gradient, color or orientation; and structural saliency which relates to combinational arrangement of individual elements. They attempt to group the salient edges by maximizing the sum of edge magnitude along the curve, while minimizing the sum of orientation difference; the result is the edge saliency network. However, Alter and Basri (Alter and Basri. 1998), in their analysis, found that the most salient location in the saliency map is not necessarily the

most perceptually salient curve. Therefore this method fails to approximate our natural visual perception.

Estrada and Jepson (Estrada and Jepson 2006) similarly used edge affinity measures of proximity and continuity within the local neighbourhood to link up edges. The initial contour may be discarded if the flanking regions demonstrate similar properties. This is checked using a colour histogram. Contour with flanking regions that have similar colour histogram are rejected; this reduces the likelihood that the selected contour will partition regions with same texture or colour.

Edge linking can also be considered as a dynamic programming problem or minimum cost graph search problem (Ren, Fowlkes et al. 2005; Ren, Fowlkes et al. 2006) by maximizing some local properties such as edge magnitude and orientation. In Bai (Bai, Yang et al. 2008), the gray level edge map is post processed using expectation maximization algorithm to group edges into longer line segments. These line segments are then linked to form contours by a heuristic that considers the weighted sum of the distance of the closest endpoints and the turning angle.

Others works use probabilistic methods to either connect the edges or suppress noisy edges. (Cox, Rehg et al. 1993; Perez, Blake et al. 2001; Felzenszwalb and McAllester 2006).

Region segmentation by means of texture and color provide another means of extracting boundaries of regions without edge detection. This could then lead to the idea of combining edge detection and texture segmentation to obtain robust contours. These are several classes of algorithms that represent images as graphs, which uses graph-theoretic techniques to partition the graphs into separate regions of similar clusters. The graph $G(V,E)$ is constructed such that the vertices V correspond to image elements (such as pixels) and the edges E is a set of links between those vertices (see Figure 3). The edges are assigned weights w_{ij} that represent how related the connected vertices are. This

relationship is called affinity between those elements and is represented by the affinity measure. The affinity value can be any of the image cues including gray level intensity, colour, texture, edge orientation and so on. The segmentation is an optimization process that seeks to globally assign subsets of nodes into homogenous regions such that region internal edges have higher weights and inter-region edges lower weights.

Wu and Leahy (Wu and Leahy 1993) proposed to model image affinity graphs as a network flow problem, and the segmentation as maximum flow or equivalently minimum cut problem, following the Ford-Fulkerson Theorem (L.R. Ford and Fulkerson 1962) on network flows. In network flow theory, the minimum cut corresponds to finding the subset of edges that, once removed, creates a situation where there can be no network flow from source to sink. In images this corresponds to splitting the image based on the weakest edge weights. In order to partition the image, the image source and sink have to be specified, and the partition will be based on the affinity to these special source and sink nodes. However, without special knowledge on the location of source and sink, the partitioning is usually not satisfying, that is, it is not capable of a good figure-ground separation.

There are several issues with segmentation using regions of similarity. The most obvious is that a single object may have multiple regions that are similar. The problem then becomes how to combine those regions into a coherent object. With this in mind, Boykov and Jolly (Boykov and Jolly 2001), introduced an interactive approach where the source and sink is user defined. Estrada (Estrada 2005) proposed a solution without user intervention in his Spectral-Embedding Min-Cut algorithm using seed regions that are identified through random walks.

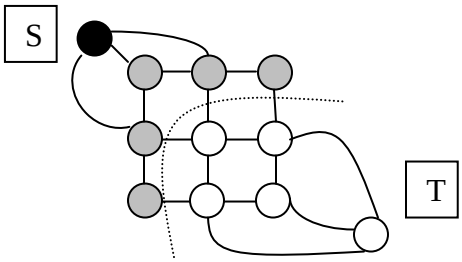


Figure 3. Representation of image pixels as graph; the source S and sink T with the dotted lines representing the min-cut that partitions the region.

The works of Malik and colleagues (Malik, Belongie et al. 1999; Malik, Belongie et al. 2001; Ren and Malik 2002) adopts the spectral approach that combines textures and edges in the normalized cut framework (Shi and Malik 2000). From the original normalized cut approach, they have extended the w_{ij} affinity weights between pixels to include measures of affinity in textures, via "textons", and edge orientation. In (Ren and Malik 2002), they considered multiscale completion of lines by formulating it as a hidden Markov problem. The hidden Markov model considers pixel line orientation and texture group as the input feature whilst tangent and position measures as the hidden variables. Ren (Ren and Malik 2002; Ren and Malik 2002) has shown that the contours do not fit well into the first-order Markov model, therefore a first-order Markov model has limitations as a model for contours.

Ren and colleagues (Ren, Fowlkes et al. 2005; Ren, Fowlkes et al. 2005; Ren, Fowlkes et al. 2006) proposed models that integrate line continuity with texture cues. An edge map is constructed using the Canny edge detector. The edge map is then trained on a classifier that determines P_b, the probability of a pixel belonging to a boundary. The classifier is trained to predict the probability of a pixel being a boundary using local brightness, texture and color information. Pixels above a certain P_b value are termed G-edge (gradient). A constrained Delaunay triangulation (CDT) is fitted to the edge map to obtain the CDT completed edge (C-edge). A local model based on G, C and θ, the angle between edges is used to determine if the edges should form a contour or not. Extending

this approach they constructed a global model that incorporates both the local model and the frequency count of different junctions by their junction angles. This is then fitted in to a conditional random field model and solved via loopy belief propagation (Weiss 2000). It was found that the global model performs best as it is able to combine local evidence of continuity and global constraints. At around 25% recall level, the precision achieved was about 75%. Therefore it works quite well in identify contours, but it does not assign contours to objects

The graph-based approach has been extended to incorporate the use of Markov random field. The image pixels are modeled as a Markov random field (Geman and Graffigne 1986; Pitas 1988; Li 1994) (Besag 1986; Geman and Graffigne 1986; Li, Sun et al. 2004; Juan and Boykov 2006; Kohli and Torr 2007). However this, approach still requires manual seeding of background and foreground. Therefore, most of the image segmentation work does not aim towards automatic segmentation of object by shape or contours.

Once the contour has been recognized it needs to be converted to a form that can be used for comparison (recognition). Popular candidates include Fourier descriptors (Gonzales and Woods 2002) for contours and image moments (Mukundan and Ramakrishnan 1998) for shape. Recently shape context (Mori, Belongie et al. 2001; Belongie, Malik et al. 2002; Thayananthan, Stenger et al. 2003; Mori, Belongie et al. 2005) have been introduced and have obtained good results.

Many shape-based retrieval (Mori, Belongie et al. 2001; Dimov 2003; Lin, Kao et al. 2004; Yahiaoui, Herve et al. 2006) systems depend on related edge detection techniques for contour extraction described above. Much of the work concentrates on shape representation (Loncaric 1998) rather than on the intricacy of the shape extraction, and so the works are tested on images with simple backgrounds such as binary animal shapes(Mokhtarian and Abbasi 1999), trademarks (Vailaya 1996), leaf (Yahiaoui, Herve et al. 2006), or line diagrams (Mori, Belongie et al. 2001). Much of the work in shape

retrieval is the evaluation of shape representations such as Fourier descriptors (Zhang and Lu 2002), shape contexts (Mori, Belongie et al. 2001), geometric and complex moments (Mukundan and Ramakrishnan 1998); and in matching approaches (Kass, Witkin et al. 1988; Adamek and O'Connor 2003; Thayananthan, Stenger et al. 2003; Sheng and Xin 2005).

There are numerous works available that use Fourier descriptors (e.g. (Folkers and Samet 2002)). Gabor filters have been used for local shape extraction for retrieval (Manjunath and Ma 1996). A comparative study between Fourier descriptors and other features have found that the centroid-radii and turning angle method give the best results (Macanu 2007). Combination descriptors of eccentricity, compactness, convexity, rectangularity and solidity (Sarfraz and Ridha 2007) was found to be better than moment invariants (Hu 1962) when tested on a subset of the SQUID database[4]. The SQUID database have also been used to develop the Curvature Scale Space (CSS) (Mokhtarian and Abbasi 1999; Mokhtarian and Abbasi 1999) which uses three global shape parameters and the maxima of the curvature zero-crossing contours as features. CSS shape descriptor has been selected for MPEG-7 standardization. CSS is known to have problems with various convex curves (Latecki, Lakaemper et al. 2000). A new technique called Farthest Point Distance (FPD) based on Fourier descriptors was shown to be better than CSS when tested with the MPEG-7 database. Shape context descriptor (Mori, Belongie et al. 2001; Mori, Belongie et al. 2005) is an alternate approach that uses the point to point relative distribution of each point in the contour.

Bai et al (Bai, Yang et al. 2008) used MPEG-7 Shape 1 Part B dataset for their object contour recognition. The MPEG-7 database consists of only binarized shapes so do not compare with the more difficult problem of using of natural scene images. Bai et al. uses a novel method whereby detected contours (as described earlier) are first compared using 1NN (1st nearest neighbor) with a database of known contours before deciding to keep the contour parts as part of the object contour.

[4] http://www.ee.surrey.ac.uk/CVSSP/demos/css/demo.html

More recent works tend towards working with natural scene images and devising new shape or edge segment based features. Ardovini and colleagues (Ardovini, Cingque et al. 2008) have worked on identifying elephants in photographs using the shape of their ears. They used the general Hough transform (Ballard 1981) as the starting point. Lu and Latecki et al. (Lu, Latecki et al. 2009) introduced a new shape descriptor that constructed a histogram based on points of all triangles formed from the edges. Opelt et al. (Opelt, Pinz et al. 2006) used edge boundary fragments that are specifically selected from the training procedure that matched edge chains and centroids in the positive images more often than negative images. It used a boosting algorithm to create the detector.

Shotton et al. (Shotton, Blake et al. 2008) also used boundary fragments but they used chamfer distance to find the best match curve. Ravishankar et al. (Ravishankar, Jain et al. 2008) used edge segments that were detached at points of high curvature so that each segments were of low curvature. The Shape Band approach (Bai, Li et al. 2009) used a coarse-to-fine procedure for object contour detection. The Shape Band defines a radius distance from the image's edge points whereby approximate directional matching of points could be performed.

Many of the shape representation or features are created based on technical arguments. The closest work to ours is Shotton et al ((Shotton, Blake et al. 2008) which, uses contour fragments from the Canny edge detector for shape-based recognition. They have achieved very good results. Our own work has achieved comparable results.

1.3 Chapter Synopsis

In chapter 2, we develop a new representation based on the wedgelet and discrete orthogonal polynomials. This is a mixed geometric and harmonic basis representation. The wedgelet representation can be used as non-noisy edge detection for removing spurious edges and gives an initial coarse image with the edges (from wedges)

represented. We will use the Ant Colony Optimization to link up these edges to obtain the object contour. We will discuss what information we have used to obtain those contours.

Based on the experience developed in Chapter 2, we develop a new approach called Segmentation via Recognition. In this approach, we used image patches for recognition and active contours for segmentation. The limitations with this approach are also discussed.

Chapter 4 expands on the Segmentation via Recognition proof of concept model presented in Chapter 3. In this chapter we develop a new object recognition approach using edge contours, and we show that it is better than existing methods. This stage will feed into the object contour extraction stage.

Chapter 5 will discuss the object contour extraction using the earlier results from the object recognition. Again we will use the Ant Colony Optimization but now with prior knowledge of the shape from the output of the object recognition stage.

We will discuss, in Chapter 6, the neurophysiological and psychological evidence of top-down and bottom-up processing in human visual processing, and an overall model and extension based on the research we have achieved so far.

Finally, in Chapter 7 we will conclude and provide a summary to the questions we have posed here.

1.4 Research Contributions

We have introduced a new image representation approach that integrates textures and geometric structure. We have also investigated an approach based on the above representation that uses Ant Colony Optimization to extract object contours. This is published as:

- Loke, K.S. (2010). Wedgelets-based Object Contour Detection using Ant Colony Optimization, Sixth International Conference on Natural Computation, Shandong.

Parts of Chapter 2 are based on the above publication.

Preliminary research on discrete orthogonal polynomials image representation was published as:

- K.W. See, K.S. Loke, P.A. Lee and K.F. Loe (2007). "Image Reconstruction using Various Discrete Orthogonal Polynomials in Comparison with DCT."Journal of Applied Mathematics and Computation. 193(2). pp. 346-359.

A study on texture was published as:

- Loke, K.S. (2009). An Approach to Textile Recognition. In Lazinica, A. (ed.) Pattern Recognition. In-TECH publications. ISBN 978-953-7619-X-X. pp.339-360.

Preliminary studies on texture was published as:

- Cheong, M & Loke, K.S. (2008). An Approach to Texture-based Image Recognition by Deconstruction Multispectral Co-occurrence Matrices Using Tchebichef Orthogonal Polynomials. ICPR 2008.
- Cheong, M. & Loke, K.S. (2008). Textile Recognition using Tchebichef Moments of Co-occurrence Matrices. Lecture Notes in Computer Science 5226. pp. 1044-1051.
- Loke, K.S. & Cheong, M. (2009). Efficient Textile Recognition via Decomposition of Co-orccurrence Matrices. IEEE International Conference on Signal & Image Processing. Kuala Lumpur.

A new framework based on knowledge-based contour extraction was introduced, which we called Segmentation via Recognition. This is discussed in Chapter 3 and parts of that section was published as:

- Loke, K.S. & Egerton, S., (2010). Scene Understanding: A Framework for Image Segmentation via Object Recognition. Sixth International Conference on Intelligent Environments, Kuala Lumpur.

We have also developed a new object recognition method using edge-detected contours that produced improved detection rate compared other recent results. This work has been submitted for publication. A new background detection based on previous research Cheong & Loke (2008) work was developed to suppress non-object background edges.

We introduced a new object contour extraction based on our Segmentation via Recognition framework by using our new contour-based object recognition algorithm, whereby the results are used to complete the segmentation by extracting the recognized object contour. This research is in preparation for publication.

All of the publications above were published either in refereed conferences or journals.

Chapter Two: **Wedgelets-based Image Representation and Object Contour Detection**

***Abstract*—We have developed a combinational image representation using the wedgelet minimization algorithm to include orthogonal basis transforms for encoding textural regions and wedgelet for regions with regular boundaries. We show that the representation approach can be used for automatic object contour detection**.

Keywords – image representation, wedgelet, ant colony optimization, edge detection, contour detection, discrete orthogonal polynomial

2.1 Introduction

A common approach to object detection is to decompose the image into a basis representation and use its coefficients as features. Image representation via basis functions includes:

1. Harmonic basis – Fourier, Orthogonal Polynomials
2. Wavelet basis
3. Multi-dictionary basis – mix and match
4. Non-orthogonal basis – Frames
5. Geometric basis – wedgelet, beamlets, curvelets

However, many of the representation are sub-optimal for certain classes of images. For example, two-dimensional tensor product wavelet bases are known to be suboptimal for coding images of smooth varying values separated by smooth boundaries (Hartmut Fuhr 2006). Various adaptive geometry-based approaches, such as wedgelet-based

representations (amongst others), have been proposed as a remedy. While wedgelets enable efficient capturing of mainly geometric features in images, many natural images are also texture-based. We show that the two approaches can be combined by using wedgelets to code edge-based images and orthogonal transforms for non-edge images. The combined complete representation is shown to be useful for automatic object contour detection.

As noted in the earlier section, most approaches using gradient-based edge detection produces spurious edges that need to be suppressed in post processing. For large images pixel level edge detection is simply too low a level to work on, due to computational work load, and much research has been on how to manage them. The use of wedgelet skips these two basic steps and produces clean edges without post processing or using many parameter settings.

The approach also provides for a built-in separation of textures and edges as a natural extension of the theory. We would be dealing with a higher level of information and this facilitates computation efficiency and higher level hypothesis forming; since the need to deal with lower level pixel information is not necessary. The possibilities include using top-down guided information, gestalt properties or learned shape probabilities. As shown by Malik (Ren, Fowlkes et al. 2005), incorporating higher level or global information can improve the contour detection. The case for us, then, is to incorporate other higher level information such as shape and perceptual properties in a meaningful way into our basic model and improve the detection rate.

.

2.2 Background

2.2.1 Wedgelet Approximations

As an initial step, we seek an initial representation that is appropriate for our task, i.e. shape and contour based representation. A pixel based representation is commonly used

for gradient based line and edge detection. Recent developments in image representation paradigms (Führ, Demaret et al. 2006) have allowed us to explore different forms of representation, especially geometry-based representation, that will obviate the use of gradient-based edge detection. Wedgelet approximations were introduced by Donoho (Donoho and Huo 2000; Donoho and Huo 2001; Donoho and Huo 2002) for geometry based image representation.

One application for wedgelet decomposition is denoising, however, we propose to use it as a basis for contour extraction. The advantage of this approach is that we are not working on single pixels but on wedgelets.

In the wedgelet representation, the image is adaptively partitioned into disjoint quadtree sets that can be approximated by wedge shapes sets (Figure 4). Computation efficiency was improved on by the rapid wedgelet approximation (Friedrich 2005; Friedrich, Demaret et al. 2007) by reducing the number of wedges and improving on computation efficiency using the "summation trick" similar to integral images (Viola and Jones 2004). The wedgelet representation has been implemented based on (Friedrich, Demaret et al. 2007) a version of discrete Green's theorem on polygonal domain. The implementation has a speedup of 10^3 compared to previous implementation.

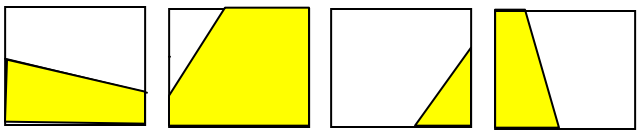


Figure 4. Examples of wedgelet. The lines that partitions the squares are the wedgelet edges.

The minimization is a huge task, but with a suitable segmentation class, e.g. wedgelet, the problem can be mitigated. First the partition search space is restricted to quadtree partitioning (Figure 5), and the local regression is limited to wedgelet family.

The problem can be separated into finding the optimal partition and the local regression model on the partition. Later, we will consider combining wedgelet and texture-based regression fitting to the dyadic squares[5].

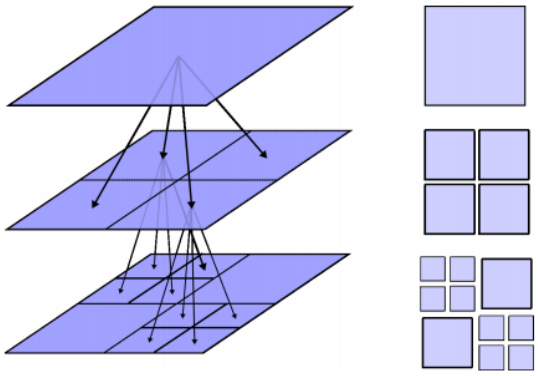


Figure 5. Quad tree partitioning.

[5] Squares that has dimensions to the power of 2.

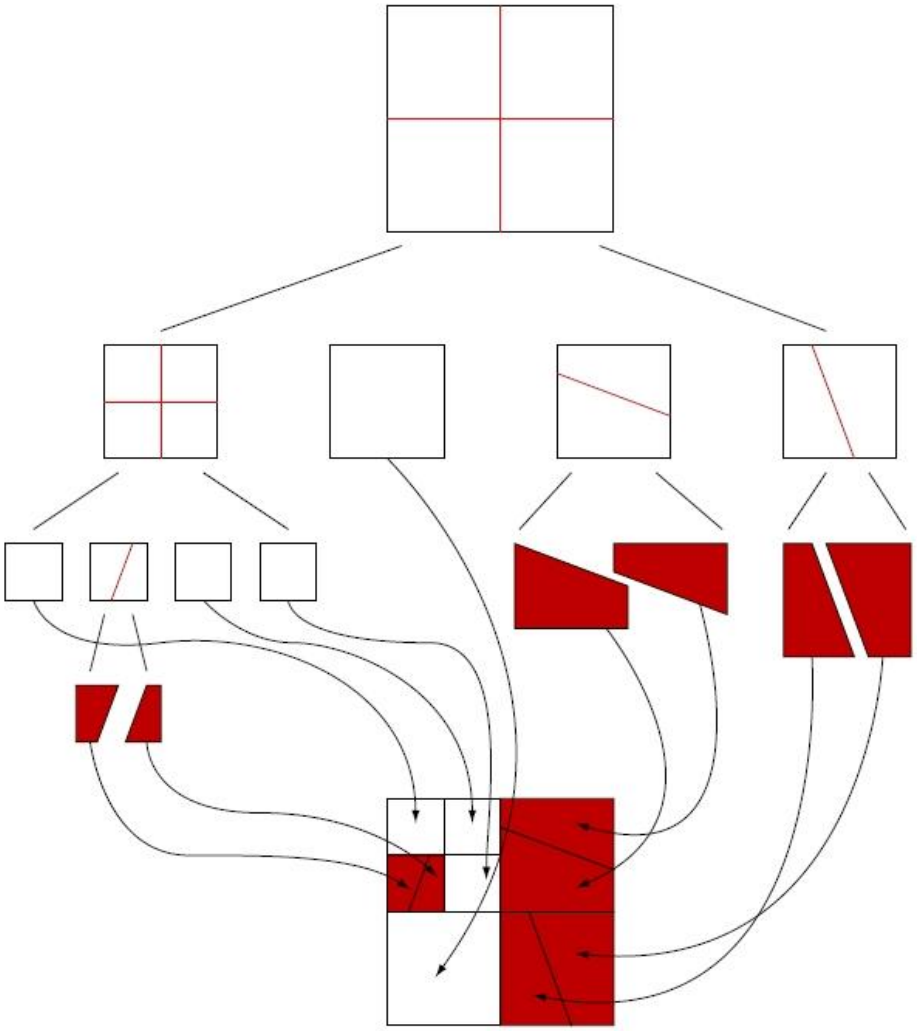


Figure 6. Example of wedgelet segmentation with quadtree partition (based on (Pongpiyapaiboon 2005))

Let the images to be elements of $\mathbb{R}^I$, where $I = \{0, \dots, 2^J - 1\} \times \{0, \dots, 2^J - 1\}$ for notation simplicity. The wedgelet decomposition can be understood as a two step procedure (Figure 6):

1. Decompose the image I into a disjoint union of wedge shape sets $I = \cup_{\omega \epsilon \mathcal{P}} \omega$
2. On each set $\omega \epsilon \mathcal{P}$, we approximate the image as a constant (within the ω wedge shape).

The wedgelet ω is selected from a fixed set $\mathcal{W}$ of wedges.

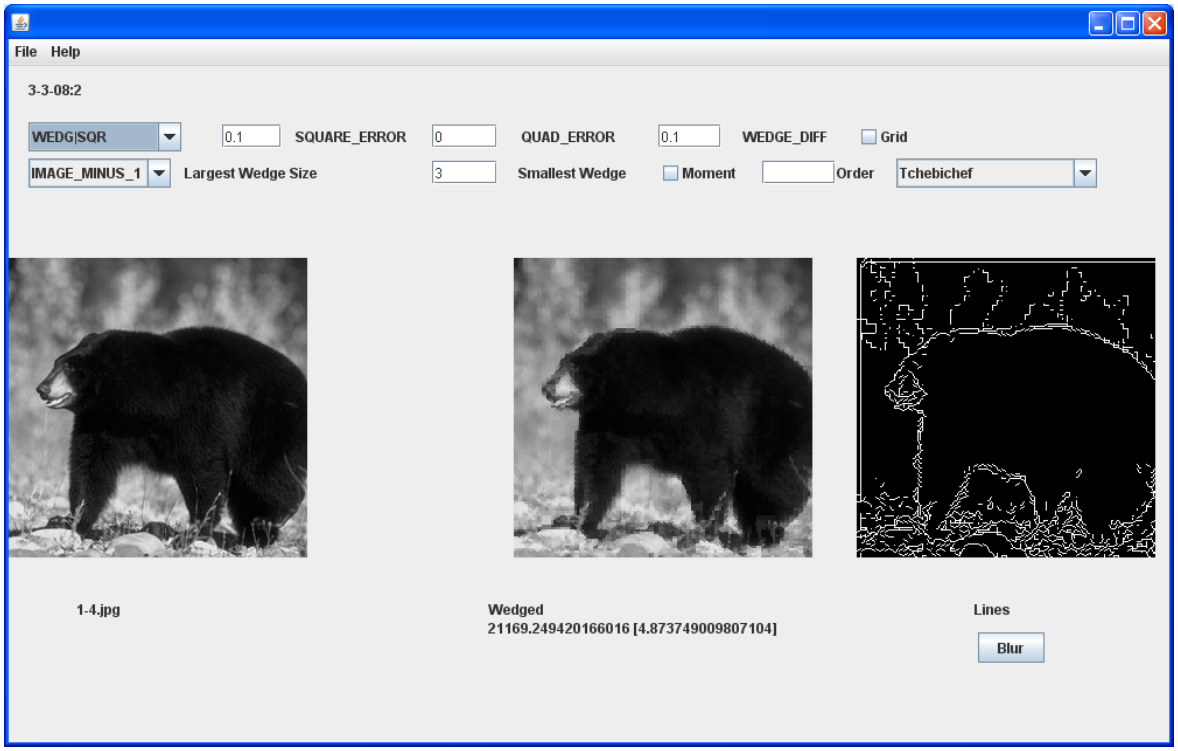


Figure 7 Example of a wedgelet representation and the resultant wedgelet edge.

A wedgelet approximation of an image *f* is obtained by minimizing the functional H:

$$H_{\gamma,f}(P, f_P) = \gamma|P| + \|f - f_P\|_2^2$$

Here f_P is a constant function for each wedge $w \in P$ such that it is the wedgelet approximation of the image at a particular partition *P*; $\| \|^2$ is the L^2 norm and γ is the regularization parameter.

The elements of *w* are obtained by splitting dyadic squares of the image along a suitable (diagonal) straight line. We want to find a partition f and wedgelet w such that:

$$\hat{w} \in \arg\min\left\{\left\| f - \hat{f}_w \right\|_2^2 + \lambda |w|\right\}$$

where

$\hat{f}_w$ the wedgelet approximation of image for the partition

$w \in W$ the set of dyadic wedgelet partition

This can be interpreted as a combination of local and global minimization. The local minimization determines the wedge split that minimizes the local approximation error$\|f - \hat{f}_P\|_2^2$ and the penalization factor. The global minimization obtained from H determines the optimal partition P from the all the local minimization obtained. In practice the partition P is constrained to a quadtree.

Figure 8. Wedgelet edges of Lena

The results in Figure 8 show the wedgelet decomposition with wedgelet edges. We call the wedgelet edge the line that partitions the decomposition squares in the image. As can be seen there are two issues to consider. One is extraneous lines that do not belong to the object, and the coarseness of the contour.

Figure 9. Wedgelet representation (right). The smallest wedgelet is 2^3x2^3.

Figure 9 shows the wedgelet decomposition of the famous cameraman image. The representation is rather coarse because the smallest wedgelet square is 8x8 pixels. Since reproduction accuracy is not what we are looking for, we have restricted the squares to be either 8x8 or 16x16.

Figure 10 shows the wedgelet decomposition in the middle and the wedgelet edges on the right. Some of the wedgelet edges are suppressed (not drawn) based on the intensity ratio between the two sides of the wedgelet. If the ratio of intensity mean between the two regions of the wedgelet is low, then the edge is not drawn. As can be seen in the sky portion, no edges are drawn as it is rather smooth. Note that there are extraneous edges in the image that do not belong to the object contour.

Figure 11 and Figure 12 shows a similar decomposition and edges, with Figure 11 using a larger smallest wedgelet of 16x16 pixels square, and Figure 12 using 8x8 pixels square. Some edges are not drawn because the intensity ratio is below a preselected threshold. In addition to wedgelets we also use full squares to model areas where the intensity is near constant within the region. In this case this is shown as a horizontal or vertical line. The line is drawn where the intensity difference is greatest.

Figure 13 shows the second stage processing of the wedgelet edges of Figure 11 by simply clustering the pixels within the wedgelet decomposition squares. This stage attempts to trace more accurate edges that are within wedgelet squares by clustering the pixels into two regions. The clustering is performed at the wedgelet square level and not at the entire image.

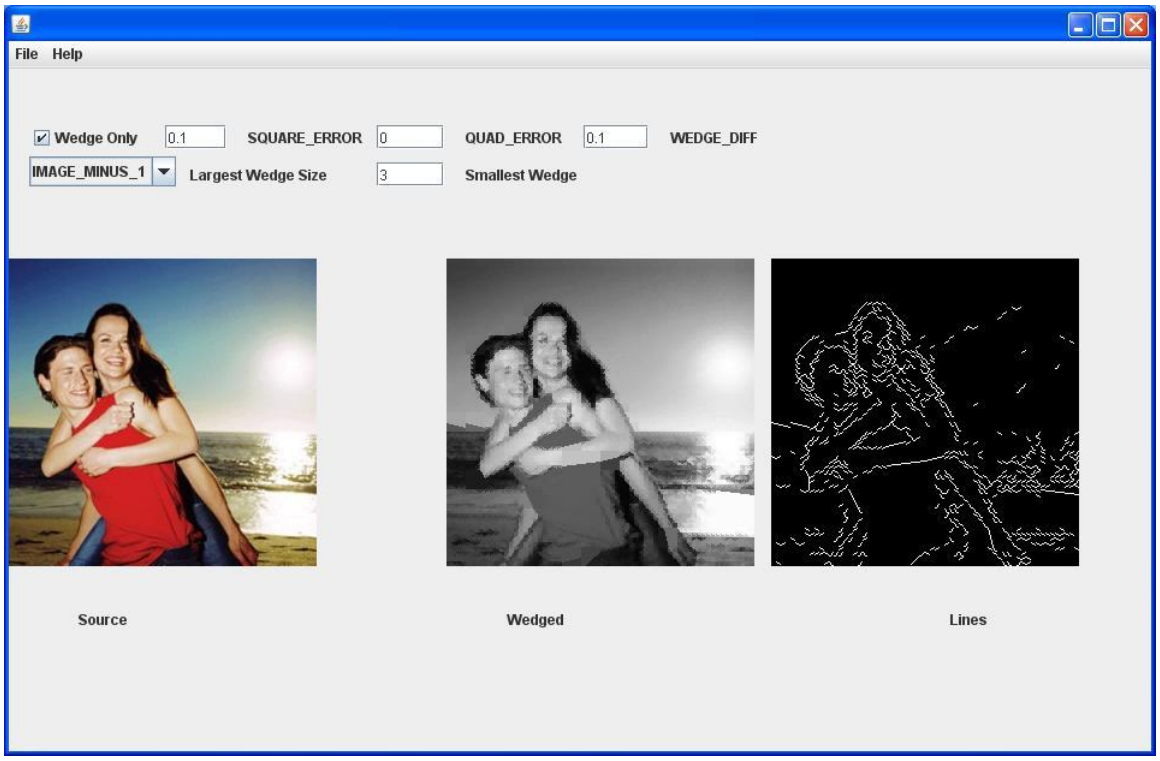


Figure 10. Wedgelet decomposition and wedgelet edges with the smallest wedge 2^3 square.

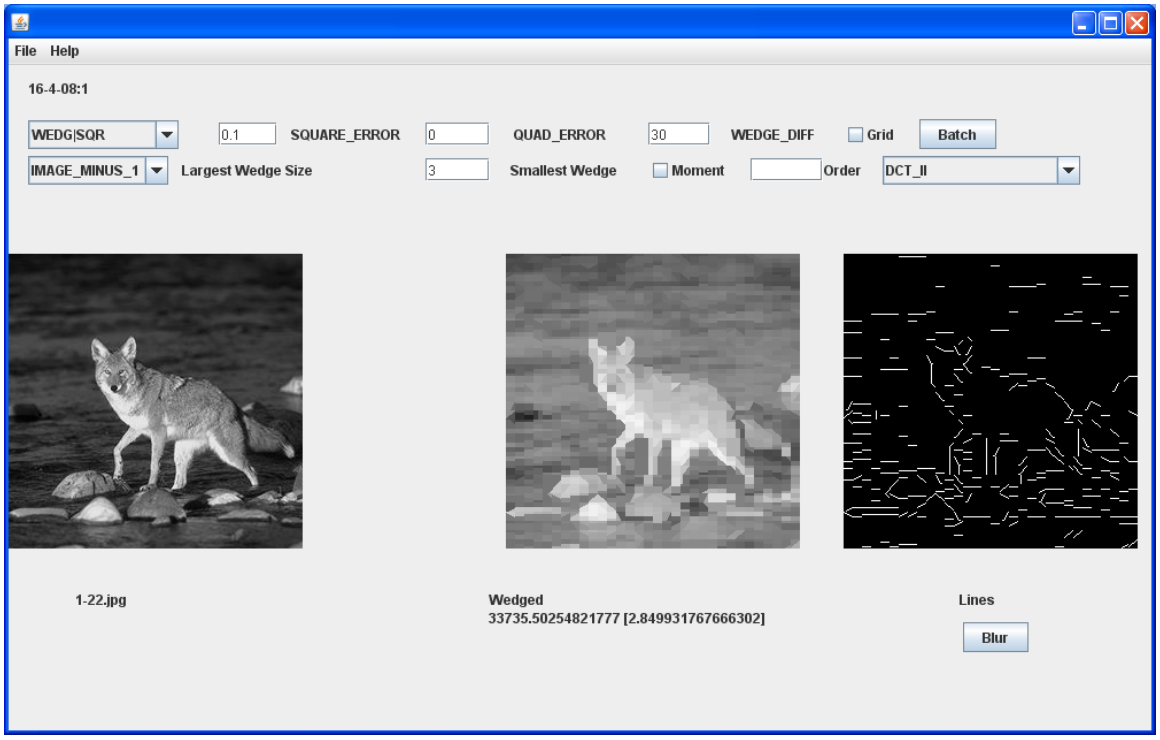


Figure 11. Wedgelet decomposition with smallest wedge size at 2^4 square.

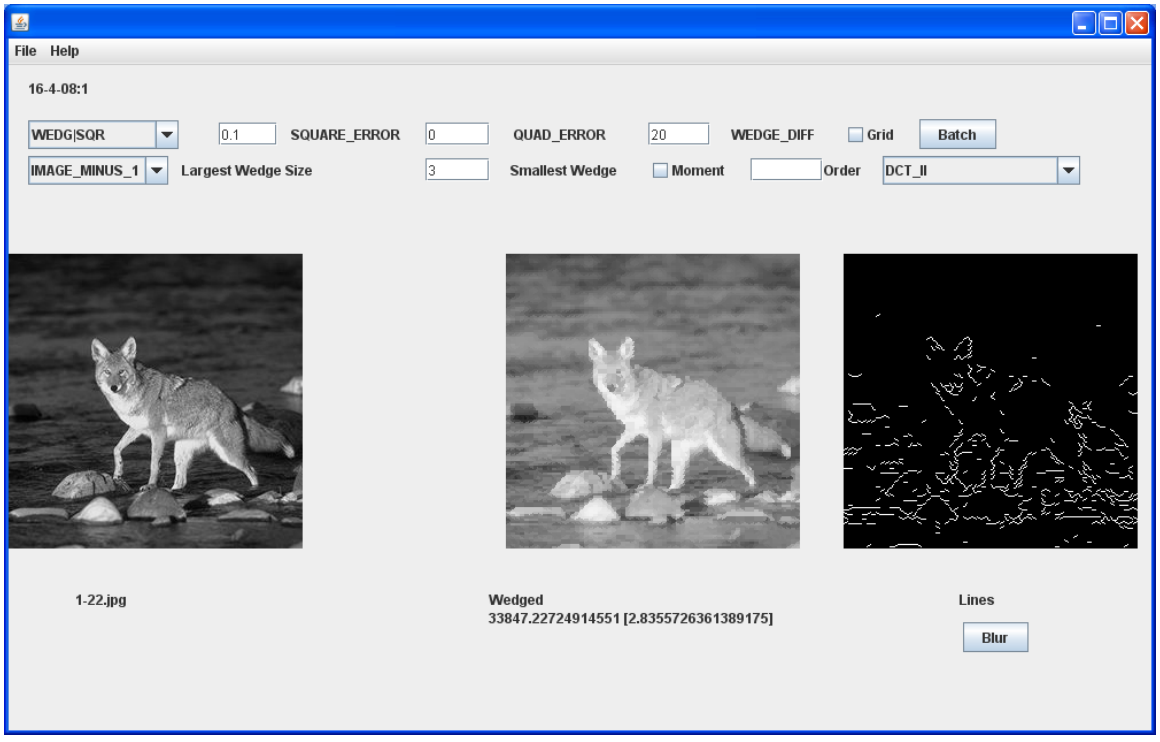


Figure 12. Wedgelet decomposition with smallest wedge size at 2^3 square.

Figure 13. Second stage edge by clustering after initial wedgelet decomposition (from Figure 11).

.

Figure 14. Quad tree decomposition of wedgelet.

Figure 14 and Figure 16 shows the various wedgelet block sizes obtained through the quadtree decomposition algorithm. The quadtree decomposition chooses the appropriate size based on the cost of representing the patch as a single larger square or as four smaller dyadic squares. Generally, constant image patches will be represented by large wedgelet

squares, as can be seen by the 32x32 pixel square in the bottom right edge of Figure 14. The smallest squares are 8x8 pixels.

2.2.2 Discrete Orthogonal Polynomial

The use of orthogonal polynomials in the field of image processing has been diverse since its introduction (Teague 1980), especially in pattern recognition (Lo and Don 1989; Belkasim 1991; Papademetriou 1992), object representation (Papademetriou 1992), edge detection (L. M. Luo 1994), texture analysis (Tuceryan 1994; Cheong and Loke 2008; Cheong and Loke 2008), and data compression (Hsu 1993). They are also used for image representation because they demonstrate significant energy compaction properties, i.e. to be able to compress the energy of the input source to a more compact form via the orthogonal polynomial transform (Mukundan, Ong et al. 2001; Yap, Paramesran et al. 2003; L. Kotoulas 2005; Zhou, Shu et al. 2005; Mukundan 2006; See, Loke et al. 2007). It was found that Tchebichef (Mukundan, Ong et al. 2001), Krawtchouk (Yap, Paramesran et al. 2003) and Hahn (Zhou, Shu et al. 2005) polynomials (one of the orthogonal polynomials) are well suited for image reconstruction. Yap et al. (Yap, Paramesran et al. 2003) showed that Krawtchouk polynomials are better than Tchebichef polynomials for certain types of image reconstruction. The use of discrete orthogonal polynomials eliminates accumulated calculation errors in continuous orthogonal polynomials as it satisfies the orthogonal property exactly without numerical approximation (Mukundan, Ong et al. 2001).

To decompose an image, the discrete orthogonal polynomials (DOP) are used to transform the image pixels intensity into image moments. To reconstruct an image, the orthogonal properties of the polynomials are used in the inverse transformation to transform the image moments back to image pixels intensity.

The basis functions are used to decompose an image $f(x,y)$ into two dimensional image moment M_{pq} of orders P-Q can be obtained with the general decomposition function:

$$M_{pq} = \frac{1}{\rho(p)\rho(q)} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} m_p(x) w(x) m_q(y) w(y) f(x,y)$$

$$\text{for } 0 \le p, q, x, y \le N-1$$

where $m_n(x)$ is a set of finite discrete orthogonal polynomials with $w(x)$ the weight function, $\rho(n)$ the rho function and δ_{nm} the Kronecker Delta, and satisfying the following orthogonal property:

$$\sum_{n=0}^{N-1} m_n(x) m_m(x) w(x) = \rho(n) \delta_{nm},$$

$$\text{for } 0 \le m, n \le N-1$$

The general image reconstruction function is used to reconstruct the image intensity $f(x,y)$ from the moment orders M_{mn}:

$$f(x,y) = \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} M_{mn} m_m(x) m_n(y)$$

The particular Tchebichef polynomial can be defined as:

$$m_n(x) = t_n(x) = n! \sum_{k=0}^{n} (-1)^{n-k} \binom{N-1-k}{n-k} \binom{n+k}{n} \binom{x}{k}$$

$$\rho(n) = (2n)! \binom{N+n}{2n+1}$$

$$w(x) = 1$$

However, to avoid numerical errors, Mukundan (Mukundan, Ong et al. 2001) proposed to normalize the Tchebichef polynomial by N^n, using the following recurrence formula:

$$\tilde{t}_n(x) = \frac{(2n-1)(2x-N+1)\tilde{t}_{n-1}(x)}{nN} - \frac{(n-1)[N^2-(n-1)^2]\tilde{t}_{n-2}(x)}{nN^2}$$

$$\tilde{t}_0(x) = 1$$

$$\tilde{t}_1(x) = \frac{2x+1-N}{N}$$

$$\tilde{\rho}(n) = \left(\frac{2n-1}{2n+1}\right)\left(1-\frac{n^2}{N^2}\right)\tilde{\rho}(n-1)$$

$$\tilde{\rho}(0) = N$$

$$\tilde{w}(x) = 1$$

The discrete orthogonal polynomials that are being studied, with its associated probability distribution function, and their default parameter(s) value can be found in See (See, Loke et al. 2007).

2.2.3 Texture and Wedgelets

The wedgelet framework allows different local regression models to be fitted to the dyadic squares. In fact, a combination of wedgelet and other models can be combined. The other model can be chosen to complement wedges, such as modeling textures that would be difficult for wedgelets. For modeling textures, we have used the low-order Tchebichef moments.

We use the moment reconstruction of the dyadic square to compare with wedgelet representation, and take the representation that has the lowest reconstruction error while obeying the quad tree partitioning error requirement.

Instead of only selecting $w \in P$, we can expand our dyadic square approximations by including those from orthogonal basis transforms such as DOP and DCT encoding. The

reason this would be useful is that wedgelet approximation has a limited dictionary (Figure 15). Adding in the mix from DOP approximation would improve the wedgelet-based representation. Generally speaking, texture-based blocks would be better represented by DOP and DCT-based methods, while geometric-based image blocks would be captured more efficiently by wedgelets.

This approach, however, also has an additional implication, especially in the area of computer vision. The fact that this approach would create disjoint partitions of the image of texture and edge-based regions has natural usage in image segmentation. We can use it to encode an image into non-overlapping texture and edge-based regions.

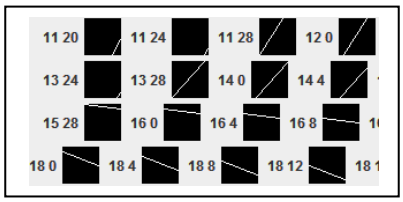


Figure 15. Example of wedgelets dictionary (showing only the wedgelet boundary).

Figure 16. Original image (left). Combined wedgelet and Tchebichef decomposition and reconstruction (middle). The quadtree partitioning grid (right).

2.3 Results of Wedgelet and Discrete Orthogonal Polynomial Representation

For our tests, we used a minimum block size of 8x8 for orthogonal functions and 4x4 for wedgelets. The reason the block size of 8x8 was chosen is because this is the standard

size used in encoding of images[6]. The wedgelet optimization algorithm will select the best size block from these minimum sizes. The size of the images is 256 x 256. In our tests we also included discrete cosine transform (DCT II) for comparison as well. The order for all the DOP functions and DCT are 4. The DOP functions we have used are Poisson-Charlier, Hahn, Krawtchouk, Meixner and Tchebichef.

The test image data that we used were Lena image and Lena+Text, which is the Lena image with text superimposed on it. The inclusion of Lena+Text tests was to see if the use of Wedgelet+DOP will be better than just DOP (and DCT) on its own. We also included 41 images containing natural images of animals and buildings in each of our test sets. (see Figure 17).

Figure 17. Examples of test images.

The results of testing with Lena and Lena+Text are shown in Figure 18. The results for the Buildings and Animals test set are in Table 1.

[6] The standard described by the Joint Photographic Experts Group (JPEG)

Figure 18. Details of W+DOP (top) and DOP reconstruction (bottom).

	Building	Animals
DOP	26.9	31.6
W+DOP	27.3	31.4

Table 1. Comparative SNR values for Buildings and Animal image test set

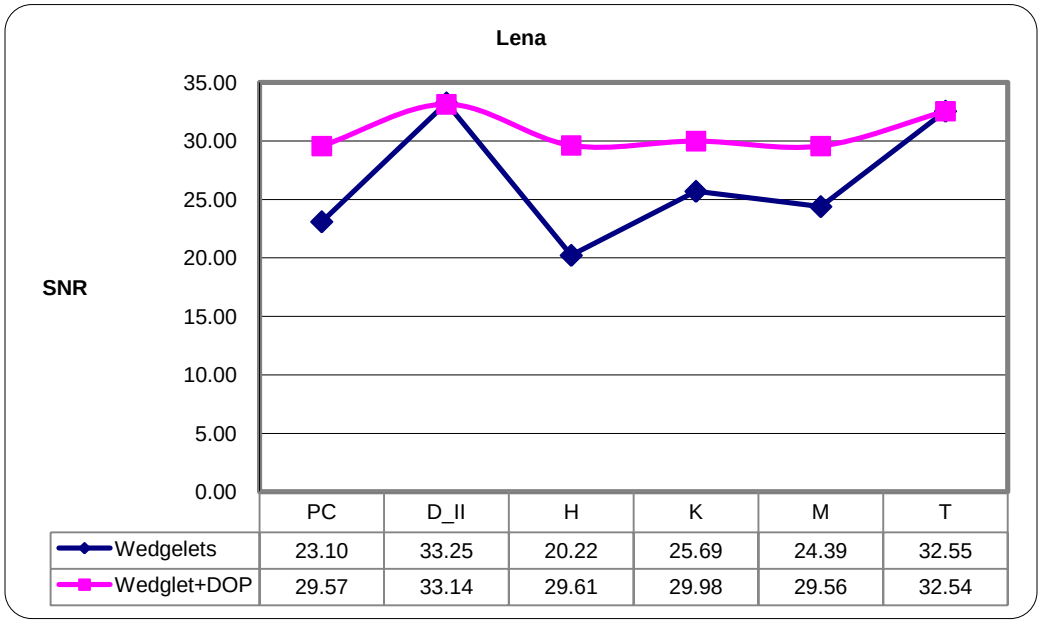


Figure 19. Comparative results of Lena

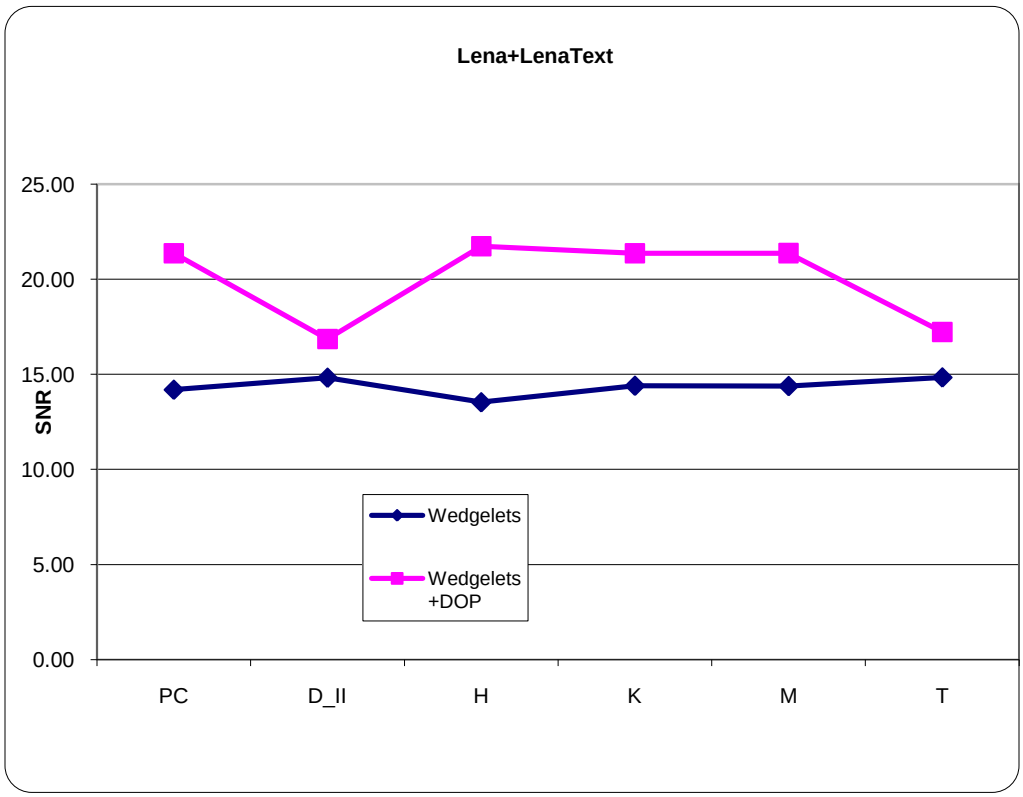


Figure 20. Comparative results of Lena+Text

The detail in Figure 18 shows the reconstruction result of applying Wedgelet+DOP against DOP alone for the straight edged section of an image. . This is to show that the top image with the straight edge has been reconstructed perfectly with wedgelet unlike using DOP. The top segment shows a perfect reconstruction with wedgelet. The reconstruction results from the Building and Animal images show that Building images have a slightly better reconstruction PSNR with wedgelets (Table 1). The reason for this is probably due the fact that images of buildings have more straight edges.

The overall results show that the Wedgelet+DOP encoding scheme can improve the results with just DOP or DCT alone. The PSNR[7] can be improved by allowing a bigger wedgelet dictionary and a smaller block size. If the straight edges were aligned, for example in the case of Lena+Text with the text aligned to segmentation boundaries, Wedgelet+DOP PSNR would increase to 29.7 whereas orthogonal transforms would

[7] Power Signal to Noise Ratio

attain 22.44 PSNR. If the alignment is perfect the reconstruction with wedgelet would be exact. These values are obtained from a previous experiment not presented here.

The overall results (Figure 19, Figure 20) show that, in general, a combined representation would yield better signal to noise ratio.

When the (DOP or DCT) basis functions have a high reconstruction factor, the combination with wedgelet does not improve, and in some cases becomes slightly inferior. However when the basis functions are rather poor, in combination with wedgelet, it can improve the reconstruction.

For example, in the case of combined encoding using DCT and Tchebichef moments, some of blocks are represented by the basis function, while a small number of blocks are by the wedgelet. In others, the majority of the blocks are represented by wedgelets while a small number by the basis functions. When the text blocks are added into the image, the reconstruction of it is improved with wedgelets.

From the results above, we have shown that Wedgelet+DOP can successful encode images. However, as it stands currently, wedgelet based representations are not competitive to other techniques. The importance of this technique is its usefulness that is amenable to further processing and analysis, as we will show.

2.4 Contour Detection

The wedgelet-based representation that we have developed above can be used for image analysis. Firstly we note that the representation is a complete and non-redundant representation. The representation is based on the decomposition of a quadtree with dyadic square tiles. Each tile stores either the wedgelet or the DOP representation. The structure of the quad-tree and the contents of each square provide non-overlapping information of the image. For example in Figure 16, the structure and the distribution of

the tiling squares (rightmost) already provide us with the information of where the object of interest is located. This, in addition to other information, can be used advantageously for image analysis.

The traditional approach to object contour detection usually begins with the edge detection phase, followed with some adaptive thresholding and edge linking. The wedgelet approach does not require edge detection as the wedgelet already provides information of the edge. The edges derived from the standard edge detection algorithm are at the pixel level. Because they are not tiled, as in our wedgelet representation, the search process for linking up the edges has to be performed at per pixel level (Lu and Chen 2008). This makes search comparatively more difficult and costly computationally compared to our wedgelet-based representation.

The non-overlapping tiling ensures that there is only one edge within a single square tile. So this suppresses noisy edges and retains only the strongest edge. This cleans up many misleading edges in other methods.

Moreover, the combined wedgelet and DOP/DCT representation provides additional information other than edges. The DOP/DCT coefficients can provide information on textures.

2.4.1 Ant Colony Optimization (ACO) on Wedgelet representation

Based on the information provided by the wedgelet representations, we used an ACO based algorithm to detect the object contour. ACO has been successfully used in Travelling Salesman (TSP) and some other problems (Dorigo and Stützle 2004; Blum 2005), and can be used in many problems where there is a construction method for the problem. Initially, artificial ants are placed randomly on the nodes of the problem graph, in this case, each node is the wedgelet tile. At each algorithm's iteration, each ant

computes the probability of moving to a new image tile not visited yet using a pseudo-random proportional rule using (1) :

$$j = \begin{cases} \arg \max_{l \in N_i^k} \{P_{il} H_{il}^{\beta}\} & \text{if } q \le q_0 \\ S & \text{otherwise} \end{cases} \qquad (1)$$

P_{ij} is the pheromone trail on connection between tile i and j and H_{ij} is the desirability of the tile j. N_i^k is the set of remaining neighboring tiles to be visited by the k^{th} ant located at tile I. β is a parameter that determines the relative importance of pheromone versus heuristic, q is a random number distributed in [0, 1] and q_0 is a parameter and $0 \le q_0 \le 1$. S is a random variable selected according to the probabilistic rule in equation (2). The parameter q_0, where $0 \le q_0 \le 1$, determines the relative importance of exploration versus exploitation and is set by the user. Therefore, the higher the value of q_0, the higher the chance those ants will choose to favor exploitation over exploration. The $p_k(r, s)$ is equation (2) gives the probability that an ant in i will choose j.

Because of the changing values of pheromone trails, the process of pheromone evaporation and reinforcement are crucial in determining which vertex an ant chooses to traverse forward to, whether or not to favor exploration over exploitation. Exploitation occurs when ants do not use a probabilistic method of selecting a path and merely selects the path with the strongest pheromone value. Exploration is the phenomenon where an ant does not necessarily always follow a trail with the strongest pheromone values (exploitation). The rationale behind exploration is to search for the existence of other solutions that may prove to be more favorable as compared to the current one.

$$p_k(i, j) = \begin{cases} \dfrac{P_{ij} \, H_{ij}^{\beta}}{\sum_{l \in N_i^k} P_{il} \, H_{il}^{\beta}} & \text{if } j \in N_i^k \\ 0 & \text{otherwise} \end{cases} \qquad (2)$$

The pheromone level is updated using the *global pheromone update* and given by:

$$P_{ij} = (1-\sigma) P_{ij} + \sigma \Delta p_{ij}^{bs} \qquad (3)$$

σ is the trail evaporation such that (1- σ) represents the pheromone persistence. This parameter is used to avoid unlimited accumulation of pheromone trails and allows the algorithm to forget bad choices made previously. ΔP_{ij}^{bs} is the pheromone quantity added to the connection (i, j) that belongs to best solution L^{bs} and given by:

$$\Delta p_{ij}^{bs} = \begin{cases} f(L^{bs}) & if\ (i,j) \text{ belongs to} \\ & \text{the best tour} \\ 0 & \text{otherwise} \end{cases} \qquad (4)$$

The $f(L^{bs})$ indicates that the change in the pheromone value is a function of the best solution L^{bs}.

Only the ant that has generated the best tour (i.e. best matched path) will be allowed to update the path pheromones. The best tour length is calculated based on the total node comparison score. The global pheromone update rule performs two tasks: (1) to perform pheromone evaporation on all model-pattern vertex pairs which do not belong on the best tour found so far, and (2) to reinforce the pheromone values of the model-pattern vertex pairs that belong in the globally best tour. This is done in order to encourage ants to search for other paths in the vicinity of the best model-pattern path found so far. Thus, the exploration process is directed towards finding possibly better solutions that exist near the best solution discovered so far.

2.4.2 Object Contour Detection Approach

The test image is first converted to 8x8 wedgelet representations. We processed the wedgelet information by removing some edges that are very weak, and edges that have edges as neighbours. The image is then clustered according to their intensity and textures based on simple statistics (e.g. mean intensity). Edges that are at the boundary of the clusters are noted as these edges tend to be real edges. In the image below (Figure 21) some edges below a fixed threshold is removed. The threshold was selected arbitrarily by visual inspection. The reduction of wedgelets is to reduce the number of paths needed to be searched by the ACO algorithm. This also reduces potential scope for errors.

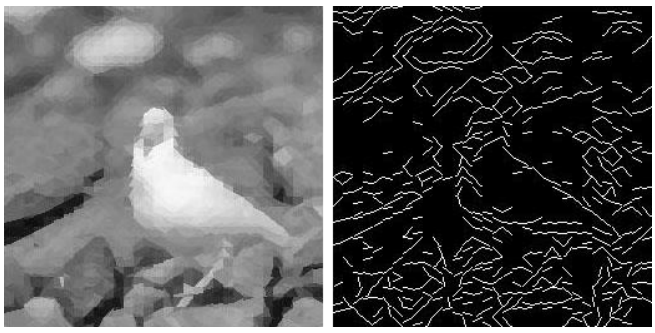


Figure 21. Wedgelet representation (left) and wedgelet edges (right).

The ACO algorithm was run in two stages, the first stage is the bootstrapping phase where the ants are to freely explore the image, the second phase is the contour seeking stage where the ants now try to return the best contour.

We modified the ACO rule (1) slightly so that the ant will have the tendency to follow the same direction when the probability of taking any direction is about equal. This can be calculated by the entropy of the choices.

$$j = \begin{cases} \arg \underset{l \in N_i^k}{\mathrm{D(l) = D(i)}} & \text{if } \mathrm{H(i)} > \tau \\ (1) & \text{otherwise} \end{cases} \qquad (5)$$

H(i) is the entropy calculated as below, *D(i)* is the direction of the contour at tile *i*.

$$H(i) = \sum_j \frac{P_{ij}\ {H_{ij}}^{\beta}}{\sum_{l \in N_i^k} P_{il}\ {H_{il}}^{\beta}} \ln \frac{P_{ij}\ {H_{ij}}^{\beta}}{\sum_{l \in N_i^k} P_{il}\ {H_{il}}^{\beta}} \qquad (6)$$

In order for the ant to return the best contour result, we need to have an evaluation function that determines the desirability of the contour. In the bootstrapping phase the results returned by the ant are ranked by L_1 and in the detection phase by L_2.

$$L_1 = \frac{\sum_{i=1}^{l} G(E_i)}{l} + \Gamma(l)$$

$$L_2 = L_1 + \Omega(\varepsilon) + P(\varepsilon)$$

where E_i is the list of tiles (wedgelet squares) that belong to the contour C, and l is the length of contour C, and ε is the list of tiles enclosed by C. The function G gives the goodness value of each E_i, and Γ is the penalization function for shorter lengths. Ω measures the circularity of the shape enclosed by C and is given by $\frac{4\pi\#\varepsilon}{l^2}$ where $\#\varepsilon$ is the count (equivalent to the area enclosed by C). P accounts for the purity of ε by counting the number of different segmented clusters within it. Basically it provides a penalization factor if there is more than one segment of clusters within the enclosed path. The choice of Ω basically limits the detection to circular-like contours and not long thin objects.

We run the ACO algorithm for each phase with global updates for best ant at the end of 20 iterations. Three different tests are performed (see Table 2).

Test No.	**Bootstrap**	**Detection**	**Remarks**
1	20 iterations, 10 epochs	20 iterations, 50 epochs	
2	20 iterations, 10 epochs	20 iterations, 50 epochs	Continue until L_2 > 0.4 regardless of iterations
3	0	20 iterations, 50 epochs	Continue until L_2 > 0.4 regardless of iterations

Table 2. Test runs and condition used.

For each test, we ran the algorithm 20 times to test for convergence, and recorded the success rate.

2.4.3 Results : Object Contour Detection

We consider the contour successfully detected if there is approximately 90% of the expected contour that is when the contour surrounds the main object. The use of less than perfect accuracy is not unusual, for example (Ferrari, Jurie et al. 2007) and (Zhu, Wang et

al. 2008) used 20% overlap of bounding box over the ground truth bounding box. We did not use a 100% matching of contour because the contours that we used are coarse pixel blocks of size 8x8 so a 100% matching is not possible. We also consider obtaining the overall shape more important; and shapes can be considered similar even if the contours do not match exactly.

Test 1 returns 85% success rate (17/20). The failure rates have very low fitness e.g. 0.08-0.09, but one with 0.32. Test 2 allows the iteration to continue, if it did not find an acceptable contour. Out of 20 runs, it returned 100% of the correct contour for each run. As there was no fixed limit of the iteration, 1 or 2 ran slightly slower, but otherwise was almost instantaneous. In Test 3, the bootstrapping phase was removed; it still performed quite well, it achieved 90% success (20/22). The same test was repeated, and resulted in 95% (19/20) success rate.

Figure 22 shows the two test images we used, with Laplacian edge-detected image on the right for comparison. Figure 23 shows a typical detected contour on the left (top and bottom) superimposed on the image scene; typical ACO pheromone trails are shown on the top-left. Bottom left shows the wedgelet edges. The contours shown are based on the wedgelet tiling. Figure 24 shows an example where the edge is drawn as a result of partition the squares by clustering into two instead of a a straight line edge of the wedgelet. Figure 25 shows various results from different ACO runs; the pixel brightness indicates the strongest path. The strength of the trail is indicated by pixel brightness.

The Laplacian edge detectors returned good edges, and could possibly be used for contour detection, but because the edges are individual pixels, the contour paths to be explored is large in contrast to our approach. Object contours are not entirely intensity gradients. The determination of whether it is an object edge as opposed to an intensity gradient requires more knowledge or information. Therefore in order to extract the object's edge, more information has to be supplied in addition to local gradients. Information such as texture, direction, intensity may be incorporated within our

framework more easily. However, determining what extra information is required is not an easy task, and has not been shown to be sufficient.

Figure 22. Images of bird and bear with their Laplacian edges (right).

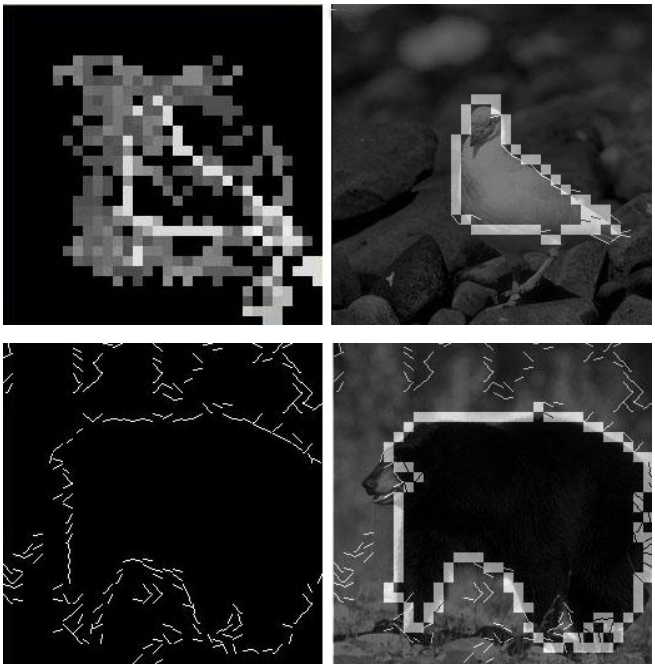


Figure 23. Typical ACO trails (top-right). Wedgelet based edges (bottom-left). Contour extracted (top-right, bottom-right).

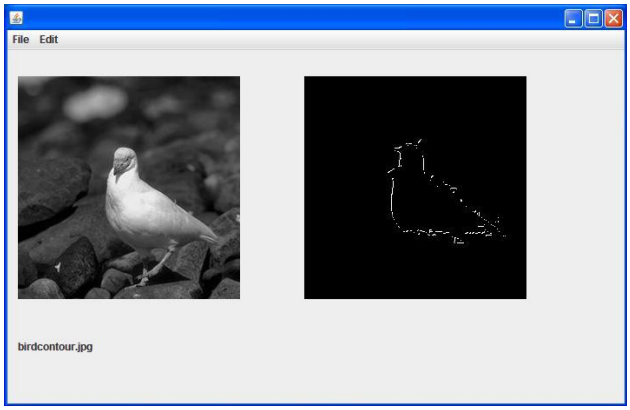


Figure 24. Sample of fine contour extraction by simple clustering

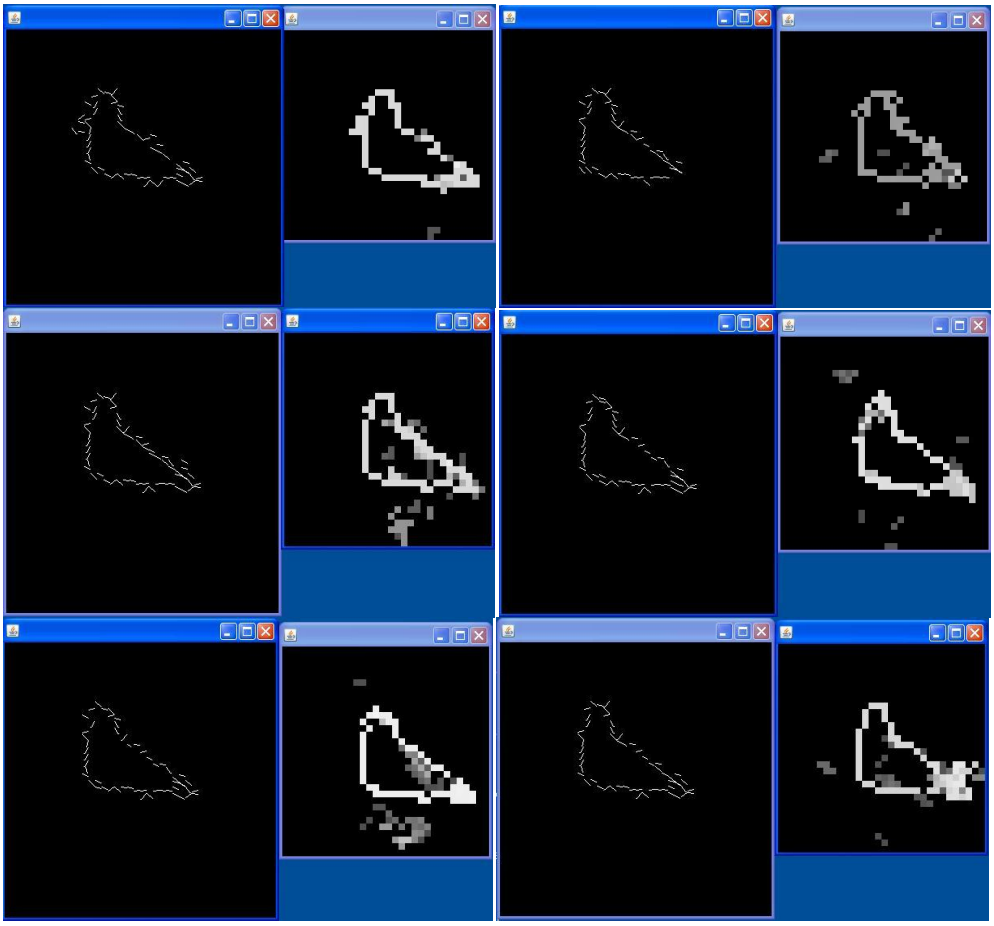


Figure 25. Samples of Ant trails and the recovered wedgelets

2.5 Discussion

We have demonstrated that the wedgelet approximation algorithm can be combined with both orthogonal basis transforms to selectively encode texture regions and wedgelet for edge boundary regions. From the wedgelet and orthogonal basis representation, we developed an automatic contour detection algorithm based on the Ant Colony Optimization algorithm. Even though we have tested it using only a limited set of images, we have developed some insights. Our experience suggest that Gestalt grouping and good continuation (Kanizsa 1979) are not robust enough to be able to determine contours in many natural situations. It seems that local properties may be necessary but not sufficient. For many years now, research has focused on local gestalt cues and no universal algorithm has been found; this suggests to us that we may have to look for a different approach. However, there may be some actual local cues that are sufficient that may be discovered in the future, so we are not making definitive claims, but rather as a guide to direct our research.

Consider the quote from Rodney Brooks (Brooks 2003):

If we see a pen lying on the desk, we can see a sharp boundary between the pen and the desk. But often when we look at the intensities of light from each little square pixel in a digital image there is no clear boundary. Pixels corresponding to parts of the pen, and parts of the desk, just 2 or 3 pixels apart, may have the same intensity values. Somehow our brain is getting a much more global understanding of what is going on, and it then perceives the boundary.

Ullman ((Ullman 1996) pg.14) wrote:

It is important not to identify object contours with intensity edges. Many intensity edges are irrelevant to the purpose of recognition, and recognition can proceed in the absence of intensity edges altogether. For example, objects can be recognized in random dot stereograms (Julesz 1971).

He also wrote (ibid, pg.1):

It does not seem, for instance, that further improvements in processes such as edge detection or stereo computation would be fundamental importance for making progress in the area of visual object recognition.

In page 163, he wrote: “Class based processing precede recognition therefore compensation can be applied first”. He cites the tests done with faces versus inverted faces under different illumination. In this case, the inverted faces returned lower recognition rate.

It seems to be the case that any improvements in edge linking and grouping with local contextual information will not provide better object contour detection, even global methods that optimizes local conditions will not be any better.

This leads us to seek an approach that can provide maximum prior knowledge rather than just local contextual information, as numerous works using bottom up cues have largely failed to provide a suitable model for segmentation and contour detection.

Chapter Three: **A Framework for Image Segmentation Via Object Recognition**

***Abstract*— An approach to image segmentation using prior knowledge is used to segment an image by first identifying objects in the image. We describe a framework that uses object recognition to determine the object and hence its contours, and use it to segment the object. The result of the object recognition is used to drive the object-based segmentation.**

Keywords- Image segmentation, image recognition, active contours, snakes, keypoint detection, visual scene understanding, content-based retrieval

3.1 Introduction

There have been many empirical studies that suggest that in visual processing prior knowledge can influence the perceptual process. This is evident in the example of the word-superiority effect when the perception of a letter in a word context is improved compared to a non-word context (Reicher 1969). Object naming (Palmer 1975) in their correct context is known to more accurate than occurring in an improper context. Peter and Gibson (Peterson and Gibson 1991) showed that shape recognition influences figure-ground organization. More directly, Vecera and Farah (Vecera and Farah 1997) have demonstrated that image segmentation can be influenced by the shape being segmented. They argue for an interactive model of image segmentation. They suggest that in the process of segmentation, partial segmentation results from bottom-up cues from Gestalt cues are sent to higher object level representations. The top-down processes will reinforce the relevant bottom up cues leading to a dynamic process that will eventual stabilize to a final segmentation. Their framework argues for a graded, dynamic interactive bottom-up top-down process. Vecera and O'Reilly (cited in Vecera and Farah 1997) conclude that bottom up cues are not always sufficient and that top-down cues on basis of familiarity can influence perceptual organization.

The empirical results from psychological research suggest that use of top-down higher knowledge based cues should be explored, especially when most work on image segmentation concentrates on bottom-up cues. We therefore propose an approach that incorporates the objects to be segmented by identifying them first. We will demonstrate the viability of the approach with a framework that incorporates appearance-based recognition with active contour deformable models for object segmentation.

3.2 Segmentation by Deformable Models

Early efforts of image segmentation have focused on edge detection and edge linking. Current work on image segmentation, other than edge detection approaches, can be divided into two mainstream approaches, namely, region-based and contour-based approaches.

Region-based approaches attempt to partition the image segment according to different image characteristics such as pixel intensity, colour and texture. Early efforts on this approach can be traced to Markov random field formulation (Geman and Graffigne 1986) and a variational approach (Shah and Mumford 1985) that allows a global objective energy function to be set up. The Markov random field encodes local properties such as pair wise similarity between two areas of interest. The energy minimization can then be solved via the minimum cut on the graph (Boykov and Funka-Lea 2006). These methods have been applied successfully to assisted image foreground/background segmentation.

Edge detection and contour-based approaches share a weakness in that the edge detection decision is made early. Graph cut and active contour methods also require an initial selection of the image segment or contour, and are sensitive to the placement of the initial contour. The requirement of an initial contour makes it difficult to be fully automated. Many approaches also require various parameters that are specified globally.

Active contours (Caselles, Kimmel et al. 1997) are one of the most sophisticated and popular contour-based approaches. It arose from earlier works on deformable models, or "snakes". Active contours, using the level set approach (Malladi, Sethian et al. 1995; Osher and Paragios 2003), improved on the weakness of the "snakes" model by allowing

topology changes in the curve evolution. The basic idea of active contour approaches is to evolve a curve subject to image-based constraints (such as intensity gradients) so as to pick out the object contour. Active contour methods can be divided into Lagrangian-PDEs (Partial Differential Equations) that govern the evolution and variational energy approach. In the variational approach, the evolution PDE is derived by minimizing an energy functional using calculus of variations.

Deformable models or Snakes were first introduced by Kass and colleagues (Kass, Witkin et al. 1988) in 1988. The basic action is a dynamic moving contour that will tend towards the boundary of the object in the image. However, it requires and initial placement of the curve. The curve moves towards minimizing certain energy that is dependent on the image intensity characteristics and the shape of the curve.

We can define the energy to be minimized as:

$$\mathrm{E}(\bar{\upsilon}) = \int_0^1 \mathrm{E}_{\mathrm{snake}}(\bar{\upsilon}(\mathrm{s}))\mathrm{ds} = \int_0^1 \mathrm{E}_{\mathrm{internal}}(\bar{\upsilon}(\mathrm{s})) + \mathrm{E}_{\mathrm{image}}(\bar{\upsilon}(\mathrm{s}))\mathrm{ds}$$

The $\mathrm{E}_{\mathrm{internal}}(\bar{\upsilon})$ term determines the properties of the contour, whereas $\mathrm{E}_{\mathrm{image}}(\bar{\upsilon})$ is an image dependent scalar potential field.

The shape of the contour $\bar{v}(s)$ is determined parametrically by the equation:

$$\bar{v}(s) = (x(s), y(s))^T$$

The $\mathrm{E}_{\mathrm{internal}}(\bar{\upsilon})$ term is the internal curve spline energy that is composed of a first-order term and a second-order term. The first term of integral $\mathrm{E}_{\mathrm{internal}}(\bar{\upsilon})$ controls the tension, i.e. how close the points of the curve, of the resultant contour, whereas the second term relates to the rigidity of the contour.

$$\mathrm{E}_{\mathrm{internal}}(\bar{\upsilon}) = \frac{1}{2}\int_0^1 \omega_1(s)\left|\frac{\partial \bar{v}}{\partial s}\right|^2 + \omega_2(s)\left|\frac{\partial^2 \bar{v}}{\partial s^2}\right|^2 ds$$

The image dependent scalar potential $P(\bar{v})$ can be written as dependent on the image intensity gradient:

$$E_{image}(\bar{v}) = \int_0^1 \rho(\bar{v}(s))ds$$

$$\rho(x,y) = -c|\nabla[G_\sigma * I(x,y)]|$$

The constant, c, controls the magnitude of the potential, ∇ is the gradient operator, G_σ is the Gaussian with spread of σ and $I(x,y)$ is the image intensity value at coordinates (x,y).

Putting it all together,

$$E(v) = \omega_1 \int_0^1 |v'(s)|^2 ds + \omega_2 \int_0^1 |v''(s)|^2 ds - c \int_0^1 |\nabla[G_\sigma * I(x,y)]| ds$$

The energy functional can be decomposed into two components; ω_1 and ω_2 components controls properties of the curve and the last integral attracts the curve towards regions of high intensity gradients which may be where the object boundary is. The first term, which is the first derivative, tries to minimize curve length by keeping the points at equal distances. If the contour is approximated by N points, $p_1, p_2, .., p_N$ in discrete case it can be written as:

$$\|p_i - p_{i-1}\|^2$$

The second term (the second derivative) ensures smoothness by penalizing high curvatures, and can be approximated by

$$\|p_{i-1} - 2p_i + p_{i+1}\|^2$$

The first expression in $E(v)$ above may be termed as an internal force and the third as the image force.

$$E(v) = E_{int} + E_{image}$$

In this formulation, each term may be considered as a force applied to the curve, and the minimum energy solution can be found. The solution $\bar{v}(s)$ to the minimizing the energy E, can be obtain via calculus of variations, must satisfy the Euler-Lagrange equation

$$-\frac{\partial}{\partial s}\left(\omega_1\frac{\partial \bar{v}}{\partial s}\right)+\frac{\partial^2}{\partial s^2}\left(\omega_2\frac{\partial^2 \bar{v}}{\partial s^2}\right)+\nabla P\left(\bar{v}(s,t)\right)=0$$

The Euler-Lagrange equation is used to update the positions of the control points. The evolving contour is re-parameterized based on the new position; the next position is calculated until convergence.

For functionals, the solution to Euler-Lagrange equation will be the function that is stationary (minima or maxima). This is analogous to finding the solution to the first order derivative to zero of a differentiable function in normal calculus. The solutions can be solved by numerical methods.

There are a couple of weaknesses with the Snakes algorithm (Angenent et al 2006) as introduced by Terzopoulus:

1. The parameters have no obvious relation between initial contour and object boundary.
2. The contour cannot change topology, i.e. the curve cannot split into two, or merge hence cannot detect unknown number of objects
3. It is highly dependent on the initial curve and placement

One early response to some of the weakness of the Snake model, was the introduction of an additional force called “balloon” force to direct the contour outwards when the initial curve is placed within the true contours (Cohen 1991).

Caselles (1993) offered a “reinterpretation” of the method motivated by curve evolution and not energy minimization. It has the advantage of capable of topology change so that several objects can be detected without needing any prior knowledge. This interpretation using curve evolution centres on boundary detection based on geodesic or local minimal distance computation. Casselles et al. rely on a classical theorem, the Maupertuis’

Principle of Least Action, to equate energy minimization to finding a geodesic curve in a Riemannian space. A geodesic curve is a local minimal distance path between given points. They have also taken the analogy further, linking it to Fermat's Principle. Fermat's Principle states that the path taken by a ray light between two points is the path that can be traversed in the least time. In essence, they have transformed a problem based on mass points and springs to a problem in time traversal in optics.

By setting ω_2 to zero, through considerations above, the following equation to be minimized is obtained as:

$$L_R = \int_0^{L(v)} g(|\nabla I(v(s)|)\, ds$$

The function $g()$ is a general function of the gradient such that $g(r) \to 0$ as $r \to \infty$. The minimum can found by gradient descent or by computing the Euler-Lagrange equation to find the extrema. Therefore according to the Euler-Lagrange equations, in order to deform the curve to minimize E_{AC} , the curve should evolve in time accordingly to:

$$\frac{\partial v(t)}{\partial t} = g(I)\kappa\vec{\mathcal{N}} - (\nabla g.\vec{\mathcal{N}})\vec{\mathcal{N}}$$

The parameter κ is the Euclidean curvature and $\vec{\mathcal{N}}$is the unit inward normal. The equation determines how each point in the active contour $\bar{v}(s)$ should move in order to decrease the length of L_R. In summary, the equation represents a curve evolution that minimizes the L_R length based on the classical snakes but obtained by considering the Principle of Least Action in physics.

The evolution of the curve is easier calculated if it is embedded in a level set which naturally allows topology changes. The evolution of the curve in level sets was also proposed by Malladi et al (1995) without the geodesic minimizing arguments. This approach basically sets up the problem in a higher dimension. The initial problem set in a planar plane is embedded another implicit function that evolves in a higher dimension (in

this case the 3rd dimension); from evolving a curve to evolving a surface. The function φ is to be defined as (Sethian & Osher):

$$\varphi(p,t) = \pm d$$

Where *d* is the distance from the point *p* to the curve at the time *t=0.*The positive sign is to indicate that the point *p* is outside the closed curve, and negative is inside the curve. At any time *t*, the evolving curve corresponds to the locus of all points *p* such that $\varphi(p,t) = 0$, and that locus is called the level curve of φ, also called the zero level set (Figure 26).

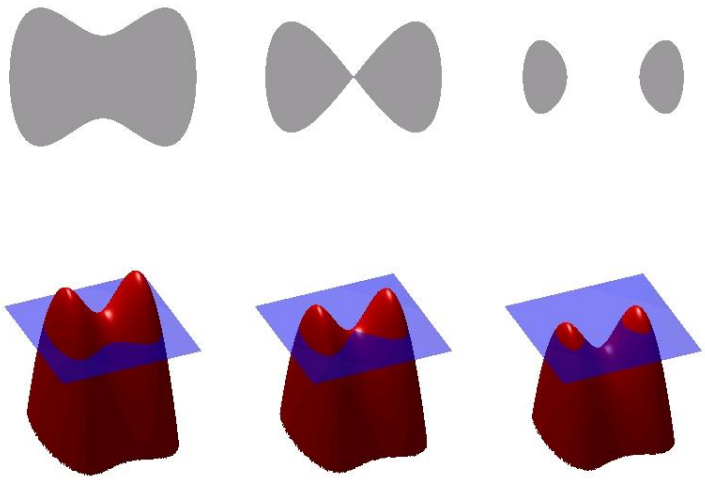


Figure 26. Illustration of Level Set Method. (source:http://en.wikipedia.org under public domain license)

For a function *p(t)* that describes the path of a point through time, the embedding function can be written as:

$$\varphi(p(t),t) = 0$$

Differentiating using the chain rule,

$$\varphi_t + \nabla\varphi.p'(t) = 0$$

If $\vec{\mathcal{N}}$ is the outward directed normal to the curve, then

$$\vec{\mathcal{N}} = \frac{\nabla\varphi}{|\nabla\varphi|}$$

Let *F*, the speed function, defined in the outward normal direction, then

$$F = p'(t).\vec{\mathcal{N}}$$

Substituting back, we obtain the Hamilton-Jacobi equation:

$$\varphi_t + F|\nabla\varphi| = 0$$

For a planar curve $\frac{\partial \mathrm{C}}{\partial t} = \mathrm{C}_t = \beta\vec{\mathcal{N}}$, the zero level set with the embedding function φ would deform according to $\frac{\partial\varphi}{\partial t} = \varphi_t = \beta|\nabla \mathrm{u}|$. Here β is the stopping function that controls how surface evolves given the image gradient. Solutions to the curve evolution in level sets can be obtained via Narrowband or Fast Marching methods (Osher and Sethian 1988; Sethian 1996; Sethian 1999)

Based on geodesic considerations above and level set embedding, Caselles et al (1993, 1995) and independently Malladi et al (1995), obtained

$$\frac{\partial\varphi}{\partial t} = g(I)(c + \kappa)|\nabla u|$$

The parameter c is a constant and $\kappa = div(\frac{\nabla u}{|\nabla \mathrm{u}|})$. The function $g(I)$ also called the stopping term because its value is close to zero at regions of high image gradient, is chosen to be:

$$g(I) = \frac{1}{1 + |\mathrm{G}_\sigma * \nabla I|^p}$$

where G is the Gaussian and *p*=1 or 2. Other decreasing functions of the gradient may be selected as well. However the curve evolution above was not robust enough to stop boundary leakage and could not pull back the front if it crossed the edge boundary. There are a couple of problems with the model. Firstly if the initial curve is not close enough to

the contour, it will not be attracted by it. And if the curve is places in an area of constant intensity it will shrink to zero.

Kichenassamy et al. (Kichenassamy, Kumar et al. 1995) and Yezzi (Yezzi 1997) proposed to add another stopping term so that:

$$\frac{\partial \varphi}{\partial t} = g(I)(c + \kappa)|\nabla u| + \nabla g.\nabla \varphi$$

The term, $\nabla g.\nabla \varphi$, denotes the projection of the gradient of g on the normal of the expanding front. This term attracts the curve to the boundaries of the objects because ∇g points towards the middle of boundaries. This new force increases the attraction of the curve towards boundaries of high gradient values.

Xu and Prince (Xu and Prince 1998) proposed a solution to the problem of concave boundaries by diffusing the attractive force. The proposed gradient vector flow field increases the capture range. In addition to above, various other additional forces have been suggested; see review in Suri et al. (Suri, Liu et al. 2002).

Even with all the additional forces, it still requires an initial placement. Incorrect placement will render it incapable of venturing into the gradient capture zone. This drawback can be partially negated by starting with multiple contours and allowing them to merge (Paragios 2005). A more problematic issue is when there are gaps in the boundaries, or when the boundary is weak, or when the background is noisy.

Some of these can be alleviated by using region-based forces (Paragios and Deriche 1999; Chan and Vese 2001), textures (Brox, Rousson et al. 2003; Rousson, Brox et al. 2003), and shape priors (Cremers, Rousson et al. 2007). The use of shape priors introduced the notion of specific contours into the segmentation and hence allows the segmentation to address the problems of occlusion, missing parts, and low contrast boundaries. With the introduction of a shape prior, the method can segment an object whose shape is similar to the given shape prior. The approach usually incorporates a

linear combination of 2 functionals; one is the active contour functional and the other functional represents the shape prior.

Leventon (Leventon, Grimson et al. 2000) introduced a model that incorporates the statistical distribution of the shape information, built from a training set of similar shapes, into the active contour model. The contour is evolved together with the λ_2 shape term:

$$\frac{\partial \varphi}{\partial t} = \lambda_1(g(I)(c + \kappa)|\nabla u| + \nabla g.\nabla \varphi) + \lambda_2(\varphi^* - \varphi)$$

The surface φ^* is the estimated maximum a *posteriori* final shape. They show results using some images from computed tomography (CT) or magnetic resonance (MR) images.

Cremers et al. (Cremers, Schnorr et al. 2000; Cremers, Schnorr et al. 2001; Cremers, Tischhauser et al. 2002; Cremers, Kohlberger et al. 2003) combined two previously separated approaches into a single framework which was called Diffusion-Snakes. The work combined the variational approach of the Mumford-Shah (Mumford and Shah 1985; Shah and Mumford 1985) framework with Active Contour framework. They propose to extend the energy functional to:

$$E(v) = E_{M-S}(v) + \alpha E_C(v)$$

The total energy $E(v)$ is a combination of Mumford-Shah diffusion energy $E_{M-S}(v)$, which measures the goodness of the segmentation and $E_C(v)$ the contour energy which accounts for the learned shape contour. In order to minimise the total energy they obtained two curve evolution equations separately for each of the terms, then they evolve it alternately. Their results show that the segmentation combines knowledge-based prior shape and data-driven image-based information. They have demonstrated their results with relatively small sample of images including some toy images, and images of hand held in front of a window.

Bresson et al (Bresson, Vanderghyenst et al. 2006) extended it to include the Leventon (Leventon, Grimson et al. 2000) shape model into the Mumford-Shah framework.

3.3 Image-patch based Recognition

The use of image-patch based local descriptors has been popular in recent years. Consumer products such as digital cameras routinely use it for face detection. Local descriptors are measurements of features taken from a small image at various location in the image. A local feature, sometimes also called interest point or region, is a distinctive image pattern of usually, intensity, colour or texture in a small image patch that is different from its immediate neighbour (Tuytelaars and Mikolajczyk 2007). The goal is to have a set of stable and repeatable measurements across scale and rotation.

The Harris detector, proposed by Harris and Stephens (Harris and Stephens 1988), is based on the second moment matrix which describes the gradient distribution in a local region around a point. A corner is detected when the two largest eigenvector from the matrix are large. The Hessian detector uses the second-order Gaussian smoothed image derivative (the Hessian matrix) calculated at multiple scales. At each scale it chooses interest points based on the Hessian matrix at that point. The interest points are those that simultaneously are extrema of both the determinant and trace of the Hessian (Mikolajczyk and Schmid 2001). This is sometimes called a blob detector because it detects blob-like structures.

Another approach uses information theory to look for salient features by measuring local complexity or unpredictability using the entropy of the local image region (Kadir and Brady 2001). Maximally Stable Extremal Regions (MSER) method looks for regions that are stable in a thresholded image under different threshold values (Mata, Chum et al. 2002). SURF (Speeded Up Robust Features) used a scale-invariant detector based on an approximated Hessian matrix using box-type filters (Bay, Ess et al. 2008). In contrast, SIFT (Scale Invariant Feature Transform), used the difference of Gaussian (DoG) at various scales for feature detection (Lowe 2004). The local extrema of the DoG images are identified as a feature or keypoints as it is called. Both SURF and SIFT prescribed their own descriptors.

The use of such detectors allowed objects to be recognized in a scene without semantic-level segmentation. However, these detectors do not often have meaningful correlation to object parts, and when they do it is only by coincidence. In order to obtain meaningful parts, these features could be interpreted as an intermediate representation, from which top-down information or external knowledge could be used to identify semantically meaningful object parts (pp. 87 (Tuytelaars and Mikolajczyk 2007))

3.4 Approach

Active contour methods, as mentioned, are sensitive to initial placement of the starting contour such that they are easily misled by local gradient. This is because it does not have a notion of an object contour and will follow the gradient regardless of the target object shape. Further research has improved on this weakness by introducing region-based features, e.g. (Chan, Sanberg et al. 2000), and the use of shape-based priors, i.e. using a predetermined shape to begin segmentation (e.g. (Cremers, Kohlberger et al. 2003)). The use of region-based features allows spatially-based features such as textures and various spatial statistics to be incorporated. All these still leaves some questions unanswered: (1) How are the initial contour and placement to be selected, and if possible, how can they be entirely automated; (2) What should the shape of the initial contour be?

New research introduced more and more knowledge, such as texture, shape priors, beyond simple intensity gradient in edge detection. The logical extension to this would be to introduce even more knowledge into the segmentation process. Taking this process to the limit would entail using the object label to be segmented in the process. Therefore, identifying the object would allow the shape of the object being segmented to be brought into the segmentation process and would not require the use of a predetermined shape priors.

The use of segmentation is for object detection and recognition by extracting its shape from its contours; i.e. we use the shape to identify the object. However, we can turn this notion on its head. Why don't we pre-identify the object, and use the knowledge to segment the image? We can perform an initial estimation of the object in the scene, and

use this information, i.e. the object shape, to segment the scene image. In turn we measure the segmentation results, if by some measure it is not good, we select another related shape, and so on. In other words we perform segmentation via recognition, or perhaps perform segmentation and recognition jointly.

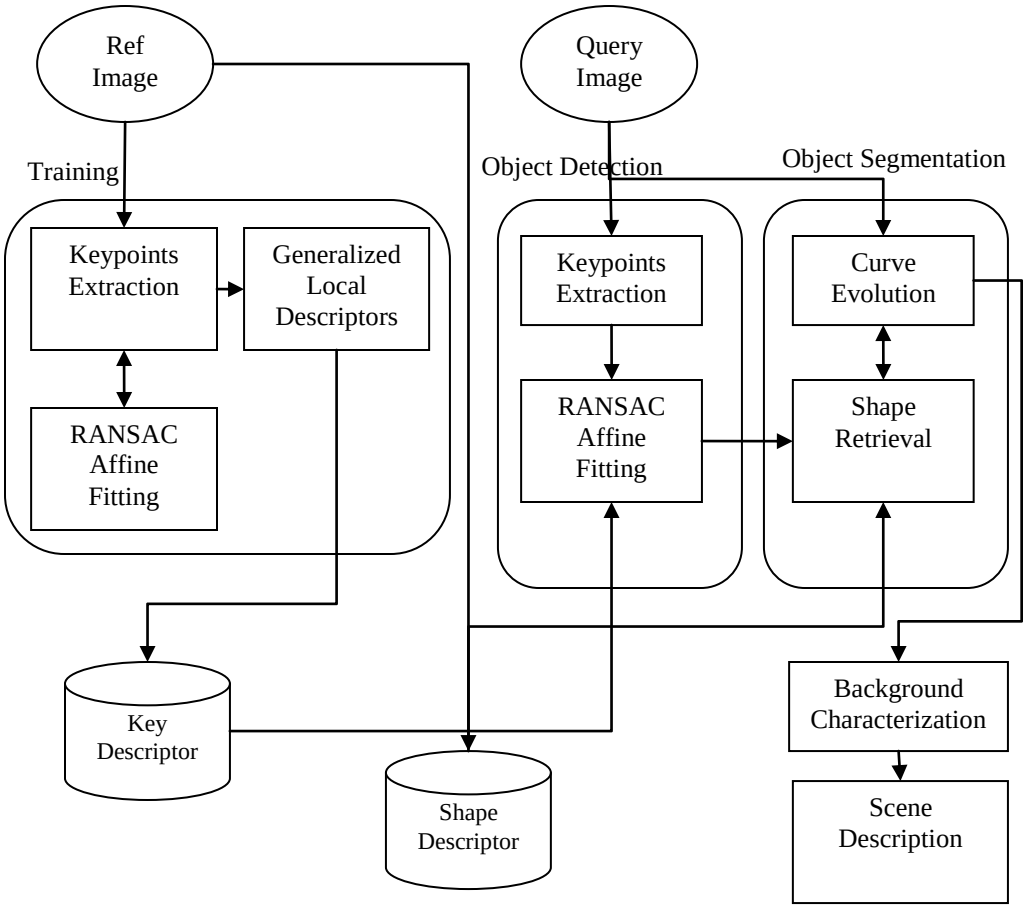


Figure 27. Framework for segmentation via recognition

3.5 The framework

In general, the framework (Figure 27) allows for a general image recognition scheme to initially identify the object in the query image, and based on that information, segment the object. Current research has shown that the use of local image descriptors for object identification is effective. However for it to be useful over a large set of object categories, it is necessary to generalize them using a large set of reference images. This process is called training. Once the object or object category has been identified, we then extract the pre-stored contours of the reference images from our database. This requires an association with each object or object category a set of predefined shapes or contours of the object. This shape, usually stored as a binary mask, will be used as input to the actual figure-ground segmentation, based on the object recognized.

The use of appearance-base local descriptors called keypoints has been successfully used in object and object category recognition (Mikolajczyk and Schmid 2005). Keypoints are measurements of image patches that are distinctive, robust to occlusion, invariant to most image transformations. The keypoint descriptors can be generalized by using a training procedure, i.e. by clustering or using principal components analysis to extract common features. The basic approach is to detect keypoints of the query image and look for the same keypoints in the reference images. In order for the object to be identified, the keypoint descriptors have to be matched. Once the matched keypoint locations are identified in the images, we need to consider that the relative keypoint locations to each other are preserved, that is the geometric relationship of each keypoint are maintained, subject to an affine transform. This implies that the geometric relationship of the keypoints between the query and reference image may only differ by a rotation, scaling or shearing.

We can consider the problem in probabilistic terms:

$$P(O_m) = \sum_j \sum_i P(O_m, r_j, y_j | k_i, x_i) P(r_j | k_i, x_i) \quad \ldots\ldots \quad (1)$$

The pair r_j,y_j denotes the j reference image keypoints and locations, while the i query image keypoint k_i is at x_i. The index i denotes all the keypoints in the query image, and j all the keypoints in the reference image m. $P(r_j|k_i, x_i)$ can be considered to be the matching quality between query and reference image keypoints. Note that the location of r_j is not required in the search for matching keypoints. The $P(O_m, r_j, y_j|k_i, x_i)$ term gives the probability of the match belonging to the object O_m from a set of m object categories for the keypoint at given location. Summing up for i and j gives the total contribution from all keypoints for $P(O_m)$. The image with the highest $P(O_m)$ gives recognized object.

In order to satisfy the image geometric constraints, the matched keypoint descriptors of both reference and query image must both be fitted via an affine transform to allow for variation due to rotation, scale, translation and shear. A RANSAC (Fischler and Bolles 1981) procedure is used to determine which points are inliers and which are to be discarded in the fitting procedure. RANSAC uses a probabilistic procedure to determine which points should belong to the match (inliers) and which should be discarded as noise (outliers).

To summarize, when a query image is presented to the system, a set of keypoints and descriptors are extracted from the image. These descriptors are compared and matched. This can be considered to be the rightmost term of (1). Then the geometric constraints are determined via RANSAC to the match with the largest number of inlier keypoints. This would correspond to the left term in (1) where position is considered. The best matched reference image will determine the shape descriptor to be used in the segmentation. This can be thought of as hypothesis generation, if segmentation fails, the next hypothesis will be considered, and so on. The detection of keypoints are quite robust under affine transformations (Mikolajczyk and Schmid 2005).

The limitation of this framework is determined by the limitation of current object recognition techniques, in particular, objects with extreme illumination, in shadow

combined with different pose and configuration (for objects with articulated sections), will be difficult to apply successfully.

A related approached is described in Leile et al. (Leibe, Leonardis et al. 2004), where the object is not actually segmented (by the object outline or contours) but given probability bounds of 25x25 patches as being in figure or ground. In their Implicit Shape Model, they create a codebook of local appearances using agglomerative clustering of Harris interest points. These will be used within the Bayesian framework to match the contribution of an image patch to the object identity. However they did not explicitly extract object contours, they just match the patches that in the object codebook, and those that was not match are assumed to be background.

Wang et al. (Wang, Shi et al. 2005) used shape context (Belongie, Malik et al. 2002) codebook, and Bayesian matching. They then verify the segmentation using bottom-up information. They call their combined top-down recognition and bottom-up segmentation. However, based on their own results, it has difficulty in providing a whole object contour, as it would segment out an object into different parts.

Whereas Leibe et al. (Leibe, Leonardis et al. 2004) and Wang et al. (Wang, Shi et al. 2005), uses the pixel probability of figure and ground, we use shape contour obtained from the recognition process to drive the segmentation. In our approach the object recognized will determine the shape and is selected automatically from possible objects recognized. It is also not necessary to train the system to learn to recognize object boundaries. Unlike the top-down bottom-up approaches, the segmentation process is independent from the recognition process that makes it easy to replace the recognition process, and does not allow the errors from the process to accumulate to segmentation. This independence can be used as a check of both recognition and segmentation.

The initial location and pose of the shape prior are also selected automatically via the affine transform matrix deduced from RANSAC. The affine transform matrix gives the geometric transform between the query and reference image.

Based on the active contour algorithm, the initial curve will evolve based on the image characteristics independent on the previous categorization stage.

The level set formulation of (geodesic) active contours require that the segmentation contour fronts, denoted by C, are represented by the zero level set φ(t,x,y):

$$C(t) = \{(x, y)|\varphi(t, x, y) = 0\} \qquad (2)$$

The curve evolution equation of the level set function on φ is determined by the level set equation:

$$\frac{\partial \varphi}{\partial t} + F|\nabla \varphi| = 0 \qquad (3)$$

The equation (2) allows us to model the segmentation as a changing process in time. The exact change, $\partial\varphi/\partial t$, is provided by (3) where it states that the change is dependent on the intensity gradient of φ scaled by F. The function F is called the speed function; it determines the speed of the evolving segmentation curve. At locations of high intensity gradient values, the curve will evolve at slow speed so that the curve will adhere and remain relatively static. This will make the curve remain at image edges because image edges lie at locations of high gradient values.

Good segmentation of objects, using active contour methods, requires that the placement of initial contour to be optimally placed. In this case, it should be shadowing the image object contour as closely as possible. This answers the two questions posed earlier in the previous section concerning the shape of the contour to be used and its placement.

The deviation from the given shape versus the segmented shape can be compared using techniques such as Fourier descriptors. If we keep the low frequency coefficients, then we can compare the rough approximate shape without the sharp details. A large deviation would indicate that it could be a false segmentation so that an alternative shape hypothesis should be considered and a re-segmentation performed. In order to deal with occlusion, the curve could be evolved using the given shape.

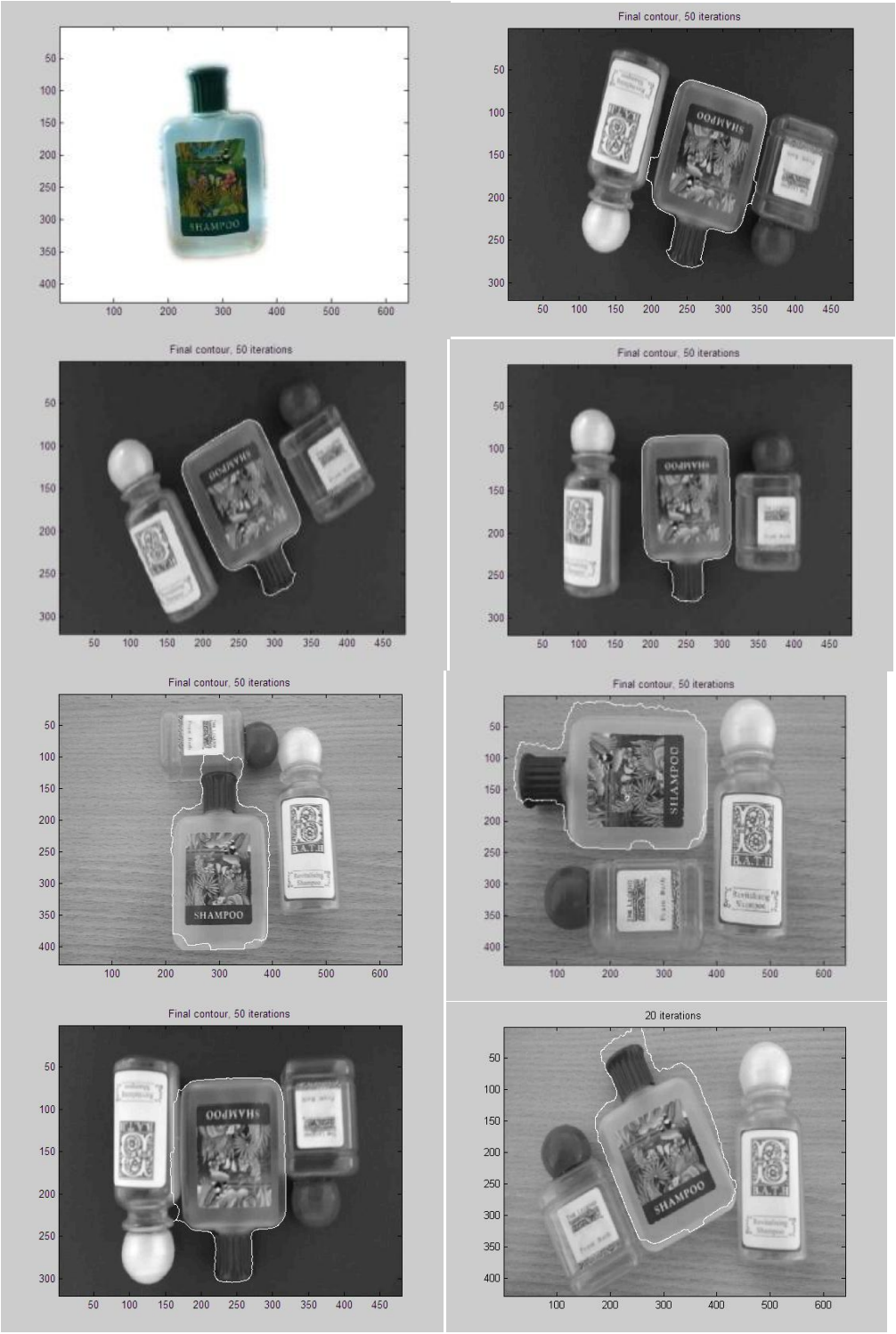


Figure 28. Segmentation results; Top-left is the object to segmented, the white curve shows the segmentation results on the rest of the images.

The successful segmentation of objects, allow the background to be characterized easily using approaches pioneered in content-based multimedia information retrieval (Datta, Joshi et al. 2008) by constructing visual signatures to identify backgrounds belonging to landscapes, sea, sky and so on. This will allow for simple scene description, and will take a step towards semantic-based retrieval.

3.6 Experimental results

We tested on a small scale test (10 objects) based on the above framework (Figure 28). The images are of various qualities, some have been shot using a mobile phone camera and others with a single lens reflex digital camera. Since the objects are distinctive, we did not employ any machine learning technique in this test.

The keypoints and descriptors are extracted using SURF (Bay, Ess et al. 2008), although other descriptors can also be used (e.g. (Lowe 2004)). As described above, we used RANSAC to find the number of inliers satisfying an RST (rotation, scale and translation) relationship instead of the full affine relationship. The ratio of inliers over matched keypoints is used to determine the object in the query image. The affine transform matrix returned from RANSAC is then used to transform the object mask of the reference image to fit the query image.

Figure 28 shows the general robustness of the results. The top-left is the target object to be identified. The rest of the images show the detected contours (as white outline) of the target object in various sizes and rotation.

We also compare the results by comparing the manual and the automated approach from the proposed method. Both approaches require the initial placement of the starting contour. Figure 29 shows, in the top-left image, the result of active contour segmentation using a manual initial outlining of the object, while top-right is the result of automatic segmentation via recognition. In the middle row, the extracted contour has been enhanced

to provide more clarity; left is the result of manual versus right, the automatic segmentation through our recognition procedure. A typical manual initial contour is seen in bottom-left, bottom right shows the automated placement of our procedure; note the tighter placement of the initial contour. In general a manual placement will not be tightly placed over the target object.

With the manual approach (Figure 29, bottom-left) as the placement is not exact, the contour tends to be snagged by the wires in the background (Figure 29, top-left, middle-left), because the high intensity gradient of the wires around the object (boomerang). In general, this will always happen with any sufficiently complex background.

The qualitative results presented provides the proof of concept; it can be seen that the segmentation is better with our automated procedure by providing the correct shape prior, pose and correct localization of the initial contour, with minimal deviation from the ideal contour (Figure 29, top-right, middle-right).

In order to compare if the segmentation is accurate, we obtained the Fourier descriptors of both the query and detected reference image. The intention was to reject the segmentation if the differences are too great. In our test, the differences were never large enough for us to reject, because the recognition stage is robust enough. Segmentation combined with recognition will provide better object and object category identification.

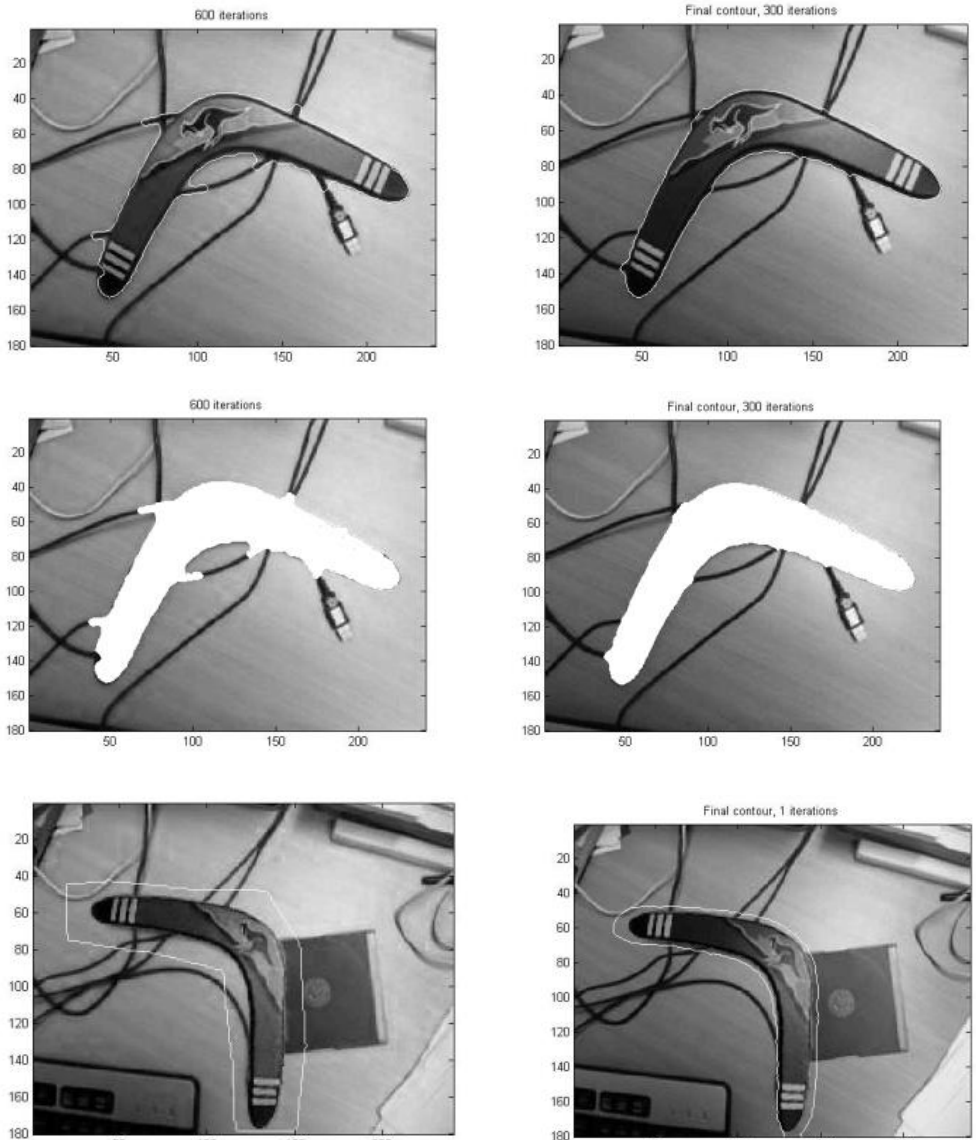


Figure 29. Manual (left column) versus Automatic (right column) segmentation. Bottom row shows the initial contour placement. Segmentation contour is the white outline in the top row, or the white object in the middle row (shown for clarity).

3.7 Conclusion

Since the experimental image set is rather small, it can be regarded as a proof of concept. Obviously a larger dataset using a standard database would be required to validate this approach. In principle, we have shown the viability of this approach.

There are also many improvements to be made, such as the use of machine learning to learn object categories. These may require specific feature detectors such as those used in face detection. The automatic selection of initial contours via recognition and the automatic incorporation of shape priors in the curve evolution may help to deal with occlusion.

Chapter Four: **Contour-based Shape Recognition**

Abstract – Psychological and physiological research on contour-based recognition are reviewed. Based on these results, we describe a new biological and psychologically motivated that could be used for shaped based object recognition. Our experiments indicate that this new feature perform as well or better than existing methods. Moreover the computation is comparatively is simpler.

Keywords – contour recognition, contour extraction, shape recognition, turning angle, curve recognition, curve saliency

4.1 Introduction – Motivation

In the previous chapter we have demonstrated the successful use of segmentation by recognition. There are two issues with the previous approach of using image patches and active contours for segmentation. Firstly not all objects are efficiently recognized by using image patches, as there are some objects that are more easily categorized by their shape or contours. For example, a horse has an easily recognized overall shape, but the colour, texture and markings of each animal may differ. Horses exhibit a diverse array of coat colors and distinctive markings (Wikipedia 2011). The Appaloosa breed, for example, has a leopard-like spotted coat, whereas a Pinto horse has a coat color that consists of large patches of white and any other color. In order to account for these differences, a large collection of image patches have to collected and trained for recognition. A simpler approach is to simply use its shape.

The second issue relates to the placement of the initial position as required by the active contour algorithm. In order to extract a clean contour, the active contour algorithm requires the initial placement to be as close to the actual object contour as possible. This is really a difficult task, and if we know where the actual contour is then the contour

extraction is redundant. Secondly, knowing the overall shape does not necessitate knowing the actual contour because the concept of shape is more general.

So the approach adopted in the previous results does not allow further generalization. We now seek a different approach, following the idea of segmentation via recognition, but based on shape. Specifically, we now seek to perform the following:

1. Determine the appropriate features that can be used for shape-based recognition
2. Determine the shape-based recognition algorithm.
3. Given the output of the shape-based recognition, extract the object contour, and segment the scene image.

The choice of the shape feature is critical, and ideally the chosen feature should be used to inform the object contour extraction algorithm.

In order to justify our choice of shape features, we need to review the physiological and psychological research in this area.

4.2 Physiological and Psychological Evidence

The primary goal of visual object recognition is to replicate the performance of the brain in performing similar tasks. The brain performs these seemingly mundane and routine tasks quickly and without conscious effort.

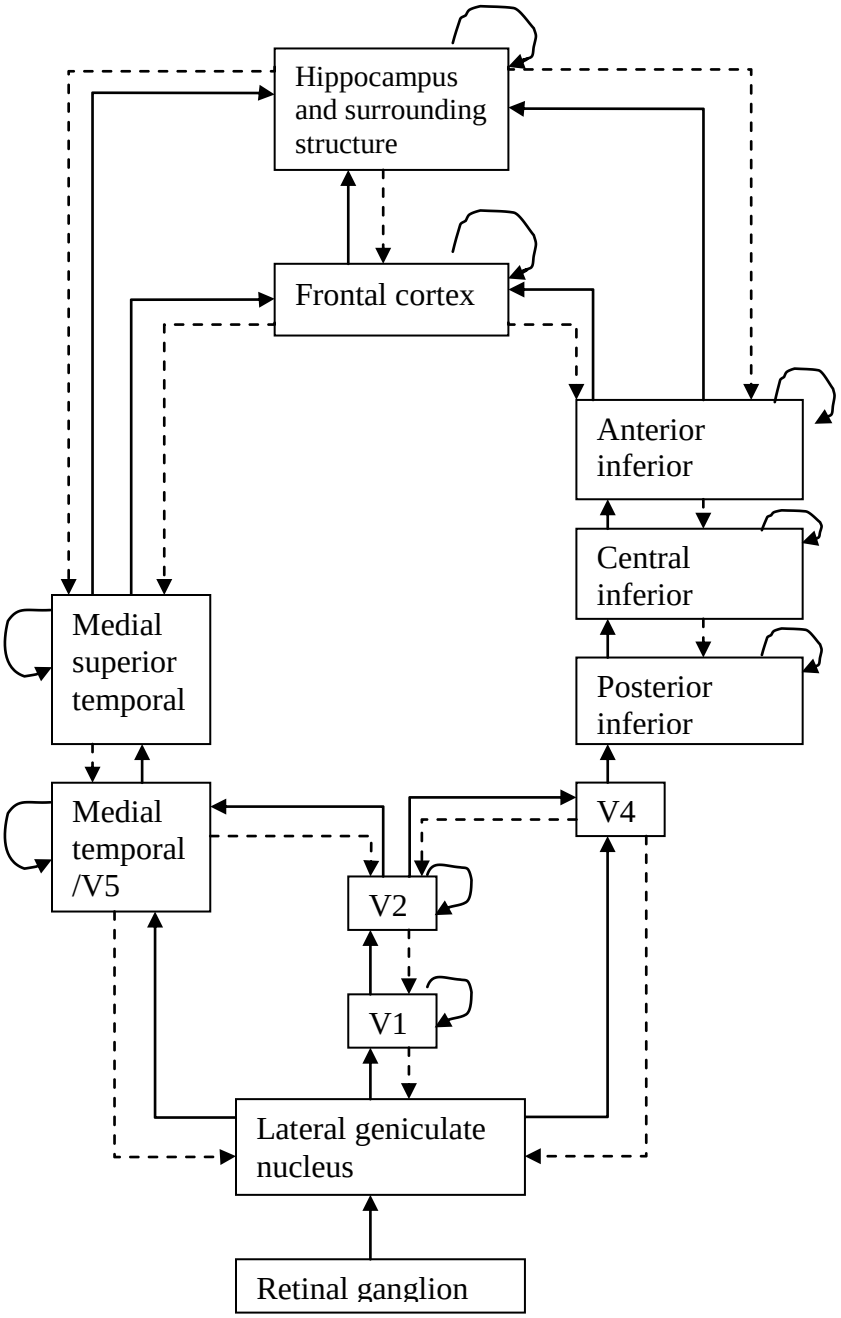


Figure 30. Schematic illustration of the basic architecture in the primate visual system (based on (Blumberg and Kreiman 2010)). Solid lines indicate forward projection, dashed lines are back projection.

The visual cortex of the brain (Figure 31), located in the occipital lobe (the back of the brain) is responsible for processing visual information. The visual cortex comprises of areas known as V1 (striate[8] cortex) and various extrastriate areas known as V2, V3, V4 and V5. The V1 areas receive information from the retina via the lateral geniculate nucleus (LGN). The V1 cortical areas forwards information (Figure 30) via two primary pathways called the dorsal and ventral stream (Ungerleider and Mishkin 1982). The dorsal stream (V1,V2, V3 to the middle temporal/MT) is associated with motion, objection locations, and control of movement, as such it is called the "where" pathway. The ventral stream, via V2, links up to V4 and the inferotemporal (inferior temporal, or IT) cortex, is associated with recognition and object representation. This is also called the "what" pathway.

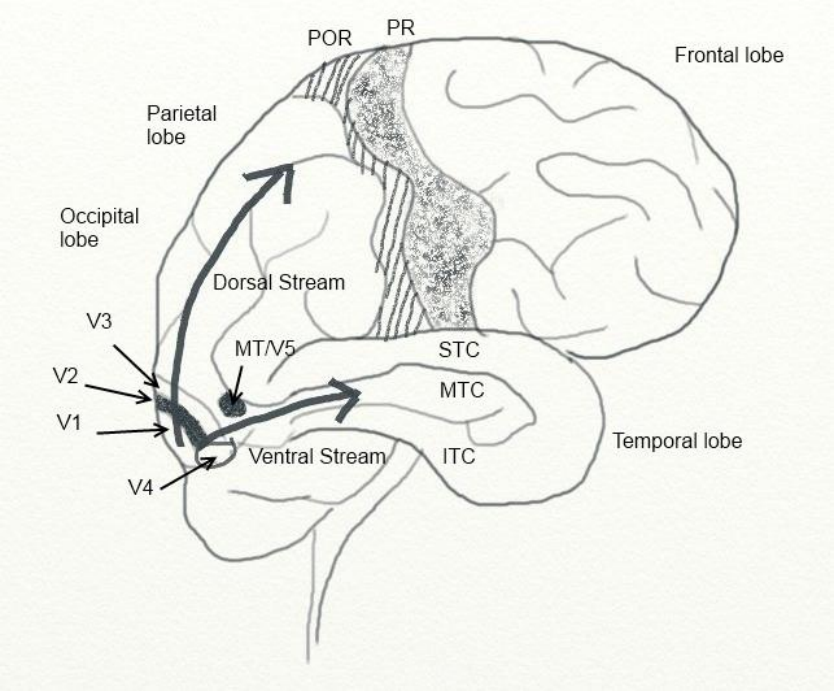


Figure 31. Schematic illustration of the human brain (based on Blumberg and Kreiman 2010)). STC: superior temporal cortex; MTC: medial temporal cortex; ITC: inferior temporal cortex; POR:post-rolandic; PR: pre-rolandic; MT: medial temporal

[8] Marked with striations; striped

The organization of cells in the V1 is retinotopic, that is, the receptive fields of the V1 neurons correspond to the spatial organization of the visual field, so that adjacent locations in the visual field are also adjacent in the receptive fields. The cells in V1 known as simple cells have small receptive fields and are sensitive to oriented contrast bars; they act as simple feature detectors (Hubel and Wiesel 1962). The so-called complex cells in V1 do not exhibit such behaviour, i.e. they have large receptive fields and are not phase dependent. It is thought that simple cells with neighbouring receptive fields feed into the same complex cell. Units in V2 are known to respond to local contour configuration including angles, arcs and intersection (Anzai, Peng et al. 2007). At the population level V2 and V4 neurons encode curvature, and shape contour (Pasupathy and Connor 1999). The posterior IT is known to integrate information on boundary fragments for complex shape coding, while the central and anterior IT encode complex object patterns (Schwarts, Desimone et al. 1983; Brincat and Connor 2006). Functional magnetic imaging (fMRI) has shown that specific visual objects stimulate the IT cortex. Studies in monkeys have indicated that the IT plays a crucial role in visual object recognition (Hung, Kreiman et al. 2005).

The connections from V1 to V2 maintain the retinotopic order, whereas the organization becomes somewhat looser between V2 and V4. In the IT, the neurons do not maintain any obvious topographic order. It seems at each forward stage, local spatial information is integrated to obtain object identity. This fits into the idea of a hierarchical feedforward model (Perrett and Oram 1993) whereby at each stage more and more object invariant properties are extracted. Finally at the last stage the object is recognized regardless of scale, translation and rotation.

Research in V4 has found that the cells respond to boundary conformation at a specific location in the stimulus, such as a certain curvature at the right, with other parts of the shape having no effect. The cells appeared to be tuned to curvature and position within their receptive fields (Pasupathy and Connor 2001). The findings suggest that at this

intermediate stage, complex objects are represented in parts as curvature position of their contour components and not the global shape. The curvature measured here is defined as rate of change in tangent angle (radians) with respect to contour length, in units of estimated receptive field radius. This is consistent with the idea of recognition by parts (Marr and Nishihara 1978; Riesenhuber and Poggio 1999).

Recognition by parts theory suggests that objects are represented by combinations of simpler components or features, which combine in a hierarchical fashion into increasingly complex representations. This theory fits into what is known. Response of oriented cells in the V1 and V2 regions are combined in V4 regions to obtain contour parts information. Cells in the later stages, such as in the IT and lateral occipital complex (LOC) may synthesize V4 signals into global shape and object identity (Grill-Spector, Kushnir et al. 1998)

Visual cortex	**Evidence of sensitivity to**
V1	Oriented bars
V2	Angle, arcs and intersections
V4 (macaque)	Curvature, colour
IT (macaque)	Object identity
LOC (human)	Object shape, identity

Table 3. Visual Cortex Functions

The human and macaque ventral streams appear to be structurally and functionally homologous up to the level of V4 (Grill-Spector, Kushnir et al. 1998). In humans, fMRI studies suggest that several areas beyond V4 might participate in object recognition. The studies discussed here suggest that a single region of cortex, the LOC, represents visual objects regardless of inducing cues, size or position. Substantial evidence from fRMI research indicates that the LOC encodes shapes of objects and not low visual features such as textures, contours, color, etc. (Kourtzi and Kanwisher 2001; Grill-Spector and

Sayres 2008), nor does it encode basic-level semantic categories (Kim, Biederman et al. 2009). Table 3 summarized some of the visual cortex functions.

In order to recognize the same object over different scales, position and various transformations, some invariant properties must be maintained over all the views. This invariance in the hierarchical model can be modeled by pooling over afferents tuned to various features at different views, sizes and location. The idea is that the cell of the high-level concept is connected to various lower level cells that activate on various local features and views to achieve view-invariance. The computational model that conforms to known anatomical and physiological constraints at that time have been demonstrated using the MAX nonlinear function as the pooling mechanism (Riesenhuber and Poggio 1999). In the model, recognition of a specific object is invariant for a range of scales and positions after training with a single view at one scale. The model however does not code the relative geometry in the image, as it uses only the presence and strength of individual features. Subsequently an updated model was developed (Serre, Wolf et al. 2007) and tested with real-world image databases. They were able to show that it performed better than existing benchmark when the features were sent to an SVM (support vector machine) or GentleBoost (Friedman, Hastie et al. 1998) classifier. This is the so-called "Standard Model" developed at MIT. The use of image patches have been explored in many computer vision algorithm such as scale invariant feature transforms better known as SIFT (Lowe 2004), histogram of oriented gradients or HoG (Dalal and Triggs 2005), and so on. However, the success of the "Standard Model" has been criticized as an artifact of the test images (Pinto, Cox et al. 2008). This is because when it is tested with a two-category problem with more variable views the result is only slightly better than chance.

While the "Standard Model" developed at MIT used image patches, another computational model based on the curvature (Pasupathy and Connor 2001) in V4 areas was developed (Cadieu, Kouh et al. 2004). Using the same feedforward hierarchical mechanism, they were able to obtain similar results from psychophysical tests on V4

areas conducted by Pasupathy (2001). The computational results obtained suggests that object centered, position specific curvature tuning of the V4 cells may arise from combinations of complex V1 cells. The results also suggest that the feedforward hierarchical network can reproduce the selectivity and invariance indicated by the V4 cells. Murphy et al. (Murphy and Finkel 2007) developed a program that has characteristics of a population of V4-like cells as described by Pasupathy (2001) and used that as intermediate representations of shape. The outputs from those V4-like responses are then fed to standard classification algorithms (e.g. SVM) using input from standard databases such as and the handwritten digits MNIST database and objects in the MPEG-7 shape silhouette database. The results show slightly inferior but comparable performance to other computer vision algorithms of Belongie (Belongie, Malik et al. 2002) and LeCun (Y.LeCun, Bottou et al. 1998). They also found the local curvature to be the most informative feature for shape recognition. Their results support the hypothesis that curvature is a robust shape descriptor.

Experiments at the perceptual level indicate that we are indeed sensitive to curvatures in contours, presumably because these cues allow us to quickly recover world structure from the image (Gibson 1950). Research has shown that in visual search task, curved contours pop-out instantly when placed among distractors of straight contours. The visual system is sensitive to these contours presumably because they are segments of the contour that contains high amount of information. In the experiments by Kristjasson and Tse (Kristjansson and Tse 2001), they conclude that the visual system is highly sensitive to curvature discontinuities, not the rate of change of curvature. They define curvature discontinuity as the point where the second derivative along an image contour is not defined or where the curvature changes abruptly. In addition, they found that curvature discontinuities need not be visible, but can be implied. They reasoned that this sensitivity is because the curvature discontinuities are particularly informative about the world structure.

Attneave (Attneave 1954) has proposed that information is concentrated in regions of high curvature of any object contour. The points along the contour where curvature reaches (curvature extrema) a local maximum contains the most information about the contour, however, not all curvature extrema points are equally salient. This can be demonstrated by removing the contours but leaving behind points that mark the regions of high curvature (see Figure 32). De Winter and Wageman (Winter and Wagemans 2008) described an experiment where they instructed participants to mark the most salient points in 260 outline shapes. The results showed that strongest curvature extrema are selected most frequently in accordance to Attneave's hypothesis, leading to the conclusion that the most salient points are located at the strongest curvature extrema.

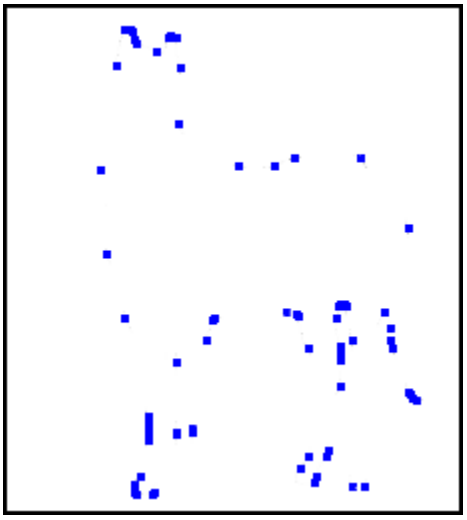


Figure 32. Turning points of an object is strongly suggestive of its shape.

Resnikoff (Resnikoff 1985) attempted to formalize the idea using the mathematical theory of Information (Shannon 1948). An improved formulation was provided by (Feldman and Singh 2005). Their derivation implied that negative curvature is more informative than points of positive curvature, however, the results from psychophysical experiments (Winter and Wagemans 2008) prove otherwise. Therefore high information content does not necessarily correlate with saliency, and is not solely determined by saliency. This can be interpreted as saying that humans do not rely on purely local

information content but integrates a more global view that includes other properties of shape. Their research also borne out that absolute curvature is not a strong factor in determining saliency as it is not stable. This view is also consistent with neurophysiological tests performed by Pasupathy and colleagues (Pasupathy and Connor 1999). Tests done on monkeys showed that cells are more sensitive to the angular orientation of the tow lines forming the curve than the curvature at the apex.

Hoffman and Singh (Hoffman and Singh 1997) proposed that the change of the normal angle from the two sides of a curve, called the turning angle, as a determinant of saliency. De Winter and Wageman (2008) concluded that the turning angle between the two flanking lines on both side of the curve is an important factor for perceptual saliency, and more so than the local curvature. The best correlations to perceptual saliency are when the normal is taken from the lines formed by neighbouring salient points (Figure 33). The strength of saliency correlates with the sharpness of the turning angle.

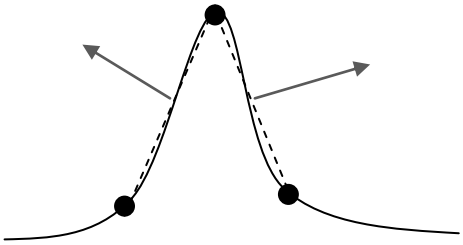


Figure 33. Turning angle measured as the difference of normal angle (arrowed line) between straight line segments connecting to two neighbouring salient points (dark circles).

Other factors of saliency have been proposed (Zusne 1970; Hoffman and Singh 1997), three of them are relative size, stick out and the inverse of compactness. Relative size refers to the area of the part that sticks out over the whole object size, and stick out is the length of the protruding part divided by the length of the base. Compactness is the area of the shape divided by the squared contour length. De Winter and Wageman (2008) reported that there are correlations with saliency for both relative size and stick out; with slightly better correlations for inverse of compactness but still less than turning angle

results. A more detailed analysis using multivariate linear regression on 10 factors, including the four discussed above and their variations, still lead to the same conclusion.

Summarizing from De Winter and Hoffman (2008), it can be concluded that not all salient points are situated at strong curvature extrema because some salient points do not occur at peaks of curvature. The strongest factor underlying perceptual saliency is the turning angle when it is measured, as the difference in normal of the adjoining lines between neighboring salient points. Saliency correlates to the sharpness of the angle. In addition, compactness, to a lesser degree, also contributes to saliency.

The results from these tests provide valuable insights for building artificial systems, and we shall be capitalizing on them.

4.3 Contour-based Object Recognition

4.3.1 Overview

The brief overview provided showed that shape and contours have psychological validity and neurophysiological correlates in the visual cortex. Therefore it makes tremendous sense to adopt those features in a computational recognition system. As psychological tests have shown that areas of high information content in the contours are sections of high curvature, it will be important to investigate the use of those features to represent curves and contours.

Our basic approach uses turning points as representation of contour fragments. The contour fragments are extracted from an edge extraction algorithm. These extracted turning points are compared with a collection of compiled exemplar image turning points. Scale and position are accounted for using a sliding window approach used in recognition. Windows of various sizes is slid across the target scene image, at each window location; the number of turning points is recorded. The window with the largest

count is considered as the probable location of the target object. The images used are scene images of various sizes consisting of trees, buildings, and so on, some with the target object, possibly obscured.

4.3.2 Related Works

A common approach is to consider the problem as a Bayesian problem. Xiafeng Ren et al. (Ren, Fowlkes et al. 2005) used constrained Delaunay triangulation to enforce curvilinear continuity. They used a training algorithm to learn probabilities; these probabilities are used in a conditional random field model via loopy belief propagation to drive edge segmentation. Their results showed clean edges, but they do not include the entire object boundary. Felzenszwalb et al. (Felzenszwalb and McAllester 2006) uses a Markov process to find salient curves and suppress noisy edges. Cox et al (Cox, Rehg et al. 1993) formulates edge-linking as a multiple hypothesis Bayesian problem using a Kalman estimator. However, they only demonstrated it on rather simple images. JetStream (Perez, Blake et al. 2001) adopts particle filtering that is based on Monte Carlo sequential importance sampling and resampling. However, the segmentation requires human interaction. In a similar vein, GrabCut (C. Rother, V. Kolmogorov et al. 2004) also performs interactive segmentation.

Lu and Latecki et al. (Lu, Latecki et al. 2009) introduced a new shape descriptor that constructs a histogram based on points of all triangles formed from the edges. For example for a point A, two other points B and C are selected from points on the contour so that a triangle is formed. Using all pairs of B and C for A, a 3D histogram is formed using the angles BAC, and distances AB and AC. Using the ETHZ shape classes, they claimed better results compared to Ferrari et al[9]. (Ferrari, Jurie et al. 2010) and Zhu et al. (Zhu, Wang et al. 2008).

[9] Their work was also published earlier as a technical report.

Opelt et al. (Opelt, Pinz et al. 2006) used edge boundary fragments that are specifically selected from the training procedure that matched edge chains and centroids in the positive images more often than negative images. It used a boosting algorithm to create the detector.

Shotton et al. (Shotton, Blake et al. 2008) also used boundary fragments but they used chamfer distance to find the best match curve. They reported the best results for the Weizmann Horse database.

Ravishankar et al (Ravishankar, Jain et al. 2008) used edge segments that are detached at points of high curvature so that each segments are of low curvature. These segments are rotated to simulate bending and scaled to create models for matching. Each of these segments is matched to the gradient image. The top 15% match of the normalized edge score is selected for further refinement. The gradient segments in the image are represented using midpoints of the low curvature segments. A dynamic programming approach is used to perform fine matching.

The Shape Band approach (Bai, Li et al. 2009) also used a coarse-to-fine procedure for object contour detection. The Shape Band defines a radius distance from the image sampled edge points from which approximate directional matching of points could be performed. Edges within the Shape Band would be then matched more accurately using Shape Context (Belongie, Malik et al. 2002).

Ferrari's (Ferrari, Tuytelaars et al. 2006; Ferrari, Jurie et al. 2010) work used a local feature they called pairs of adjacent segments (PAS). Each pair of connected segments forms one feature set that includes the mean of the two segment centres, distance between the segment centres, edge strength, a descriptor that encodes the shape of the PAS using the segments' orientations and lengths; and the relative location vector. A codebook is created by clustering the PAS inside all training bounding-boxes according to their descriptors. These will be used for matching or recognizing object shapes. They used

Hough voting to obtain rough estimates of the location of the object. From the initial location obtained, a more accurate matching using their Thin-Plate Spline Robust Point Matching algorithm (TPS-RPM) was performed. TPS-RPM matches the two point sets V and X by applying a non-rigid (affine and non-rigid warp) mapping to V. It estimates both the correspondence between V and X, and the mapping that minimizes an objective function. The objective function included the distance between points of X and their corresponding points of V after mapping them by the TPS, and the regularization terms for the affine and warped components of the TPS. They presented results on the ETHZ shapes and INRIA horses. Bounding box results are good, with detections-rates around 80% at the moderate false-positive rate of 0.4 FPPI (false positives per image) except for the giraffe images. They claimed that the lower performance on giraffes is mainly due to the difficulty of building shape models from their extremely noisy edge maps.

The last five approaches are very similar to our work, in that edge fragments are used for matching. However, the features from the curve fragments are rather arbitrarily selected without theoretical support, apart from results. Alternatively they are selected based on training discrimination tests. Features that are selected based on training tests are probably too limited because of their dependency on training examples.

In contrast to the works above, our approach is to select features that have psychological, perceptual and neurophysiological basis, i.e. we will make use of the curve's perceptually salient point, the turning angle, as the representation of the curve fragment.

4.3.3 Turning Points

In this section, the mathematical framework for obtaining the points with high turning angle is presented. Following Feldman and Singh (2005), we develop the relationship between information content and curvature. Based on Shannon theory of information (1948), information content is dependent to the negative logarithm of the probability.

$$u(M) = -\log[p(M)]$$

The quantity *u(M)* is also called the surprisal of *M*. The information content is the expected value of the surprisal.

$$I(M) = -\sum_{M} p(M)\log[p(M)]$$

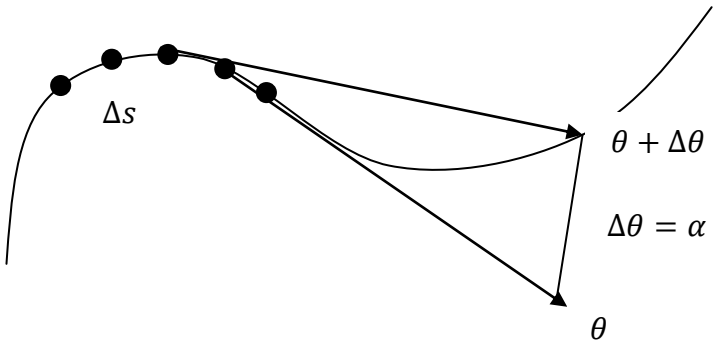


Figure 34. A curve sampled at Δs intervals. Each point has a tangent θ, the angle α or Δθ is the difference between successive tangents.

Consider now a curve of length L (Figure 34), sampled at Δs internals at *n* uniformly separate points. The angle α or $\Delta\theta$ is the angular difference as the tangent direction of the curve changes from point to point. The spread of α is modelled as the von Mises distribution, also known as circular normal distribution, is a continuous probability distribution on the circle. This distribution is used rather than a Gaussian is because the Gaussian has its support between $(-\infty, \infty)$ whereas angle changes are between $(-\pi, \pi)$. The von Mises distribution can be thought of as the circular analoque of the Gaussian

distribution. Moreover the von Mises distribution asymptotically approaches the Gaussian as the spread narrows. The distribution can be calculated from the Bessel function of order 0.

We can now obtain the surprisal as:

$$u(\alpha) = -\log\,[p(\alpha)]$$

Taking the curvature κ as the change in tangent direction as we move along the curve, we obtain

$$\kappa \approx \frac{\alpha}{\Delta s}$$

Note that this approximation becomes exact as Δs tends to zero. The von Mises distribution for α with the constant A and the spread parameter b is

$$p(\alpha) = \frac{Ae^{bcos(\alpha)}}{2\pi I_0(b)}$$

I_0 is the Bessel function of order 0.

The curvature κ is distributed likewise to α because of the approximation, we obtain a similar distribution:

$$p(\kappa) \approx A' \exp\,[b(\Delta s)^2 \cos\,(\Delta s\kappa)]$$

The surprisal is then

$$u(\kappa) \approx logA' - b(\Delta s)^2 \cos(\Delta s\kappa)$$

In other words $u(\kappa)$ is proportional to $-\cos(\Delta s\kappa)$, that is the surprisal is proportional to the negative cosine of the product of scale and curvature, and increase monotonically with scale invariant version of the curvature $\Delta s\kappa$:

$$u(\kappa) \propto \cos(\Delta s\kappa)$$

However the result from this assigned a higher surprisal to points of negative curvature, and this does not agree with psychological experiments (De Winter and Wageman, 2008). Since it disagrees with psychological experiments we will disregard the signs. Based on the location of the surprisal (see example Figure 35), we calculate the location of the turning angles. We first find the local neighbourhood peaks of the surprisal and measure the normal angles adjoining two points on either side of the central point. The largest of difference in normal within the neighbourhood we keep as the turning points of the figure.

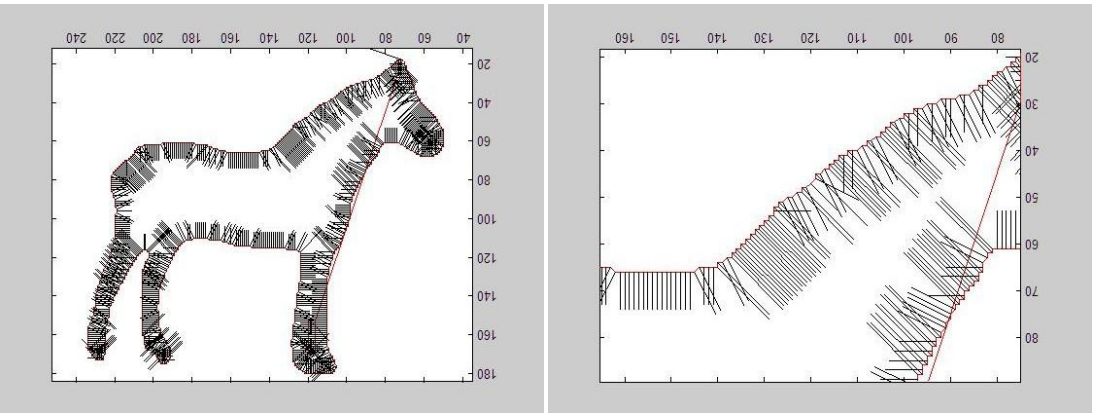


Figure 35. Surprisal location and magnitude based on formula by Feldman and Singh (2005). The length of the normal lines indicate magnitude of the surprisal.

4.4 Approach and Implementation

4.4.1 Image Preparation

The images from the database collection are separated into two groups. The first group, a smaller set, is used as exemplars for training, and the other larger group is for testing. The training images are first resized to a fixed standard size. Then from the training set images we obtain a set of perfect object contour masks (as a binary image of black background and white object). The object masks are edge-extracted using the Canny edge extraction algorithm. These edges are then used to calculate the turning points Figure 36 as described above. All the extracted turning points are then merged into a single file that

will be used for recognition comparison. The merged file consists of all the x,y coordinates of the linked turning points (Figure 37). These will be used for comparison purposes in the recognition stage.

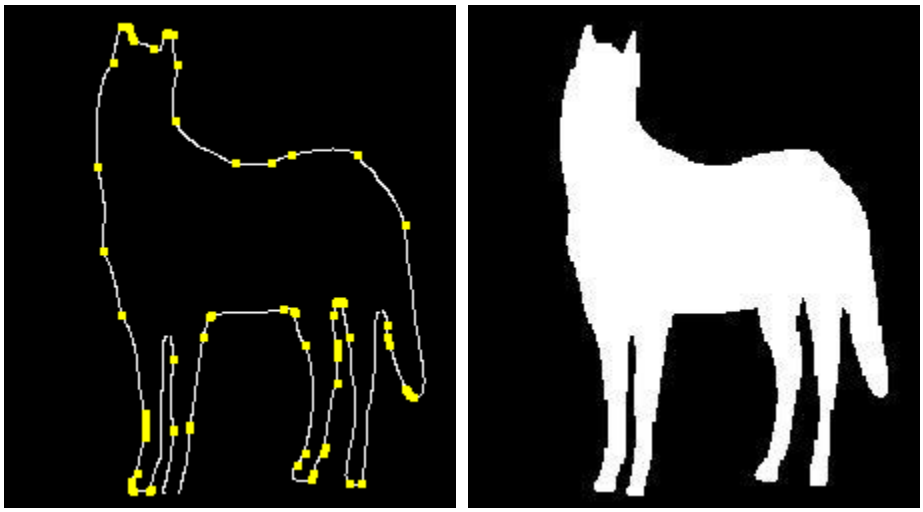


Figure 36. (left) Turning points in yellow from the object mask (right) as obtained by our program.

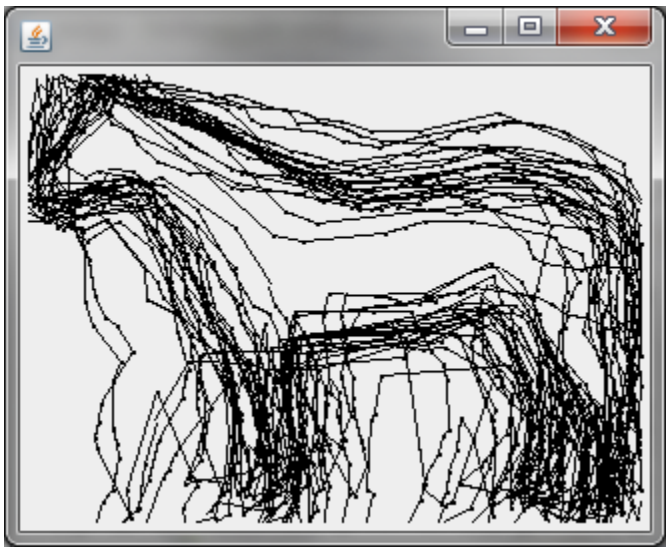


Figure 37. The exemplar image turning points used for matching

The testing images are prepared in the following steps:

1. Blurring
2. Canny edge extraction (see example Figure 38)
3. Line labelling
4. Unbranching (see example Figure 39)
5. Line tracing (see example Figure 40)

The first step, blurring, is performed using the standard Gaussian spatial filter, and then followed by the Canny edge extraction.

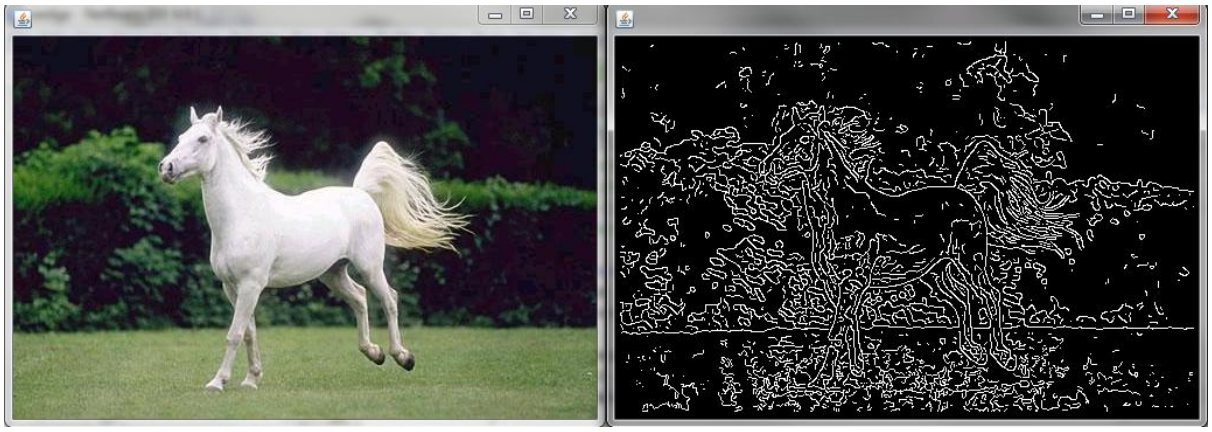


Figure 38. Image of horse118.jpg (left) and the Canny edge-detected result (right) for comparison. We only used the edge image after blurring.

Each of the marked edge points are then labeled into different edge fragments. Those fragments less than a certain length (in our case we used a fixed length of less than 15 pixels; this could be made dynamic according to image size). In order to simplify turning point extraction, loops and branches in the edge fragments are eliminated. In the case of branching edge fragment, the shorter edge fragment is disconnected from the longer branch. Loops are broken arbitrarily.

Figure 39. Example of the un-branching; the blue points shows the starting pixel of each edge fragment.

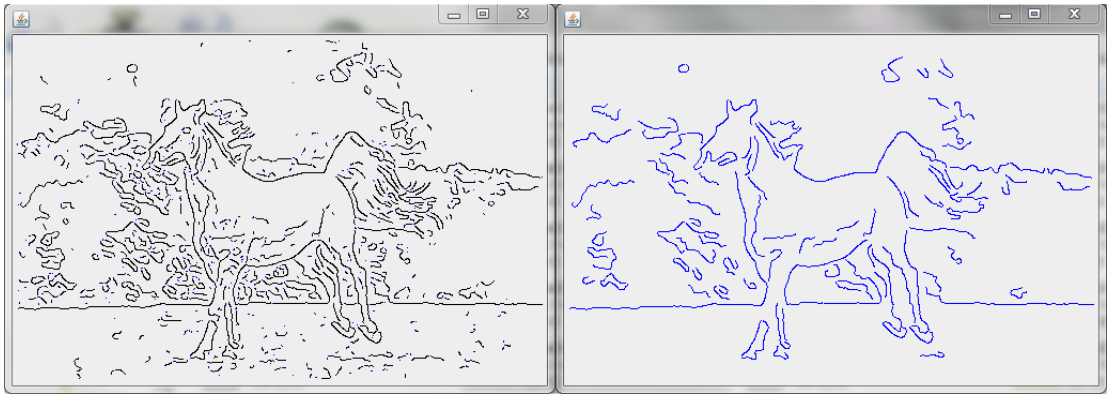


Figure 40. The edge image of horse118.jpg after blurring and un-branching (left); and the image after removing short fragments (right).

From the un-branched edge fragment, the fragment is traced to find the starting and end point. After these the edge fragment is ready for turning point extraction.

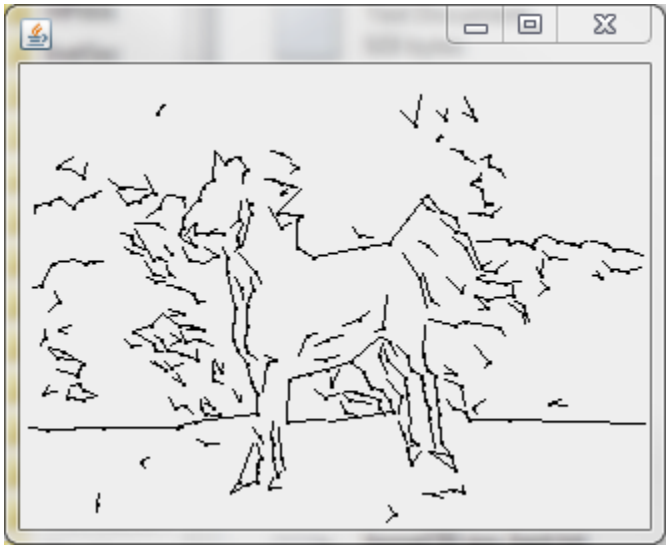


Figure 41. Image shows the extracted turning points and the straight edge linking them. These are the features used for detecting the target object in recognition.

4.4.2 Turning Points Feature Extraction

For every point $\bar{v}_i$ where the previous point vector is $\bar{v}_p$ and the next point vector is $\bar{v}_n$ we calculate the magnitude of the angle α formed by the previous and next point. In practise we take the smoothed version by averaging over the resolution size of Δs.

$$\alpha = \frac{\bar{v}_p \cdot \bar{v}_n}{|\bar{v}_p \cdot \bar{v}_n|}$$

Next we calculate the surprisal

$$surprisal_i = -\log\left[\frac{\exp\left(\cos\left(\alpha - \frac{2\pi\Delta s}{N}\right)\right)}{2\pi I_0(1)}\right]$$

Over a small local neighborhood, we mark the peak surprisal with local turning angle that exceeds a threshold within that window. If there are many equal maximum values, we pick the one point in the middle of the window; these will be our turning points.

These points are the locations of the salient points on the curve. However, on their own, do not contain adequate structural information for robust recognition in our approach. For each of these salient points we include the location and direction of the next and previous salient points connected to it. With the previous and next point we can calculate the turning angle.

In summary the features we used are:

a. Salient points from the peak surprisal with local turning angle threshold exceeded within a local window
b. Salient point to salient point turning angle at each point
c. Length and direction of the previous and next salient point

4.4.3 Feature Matching

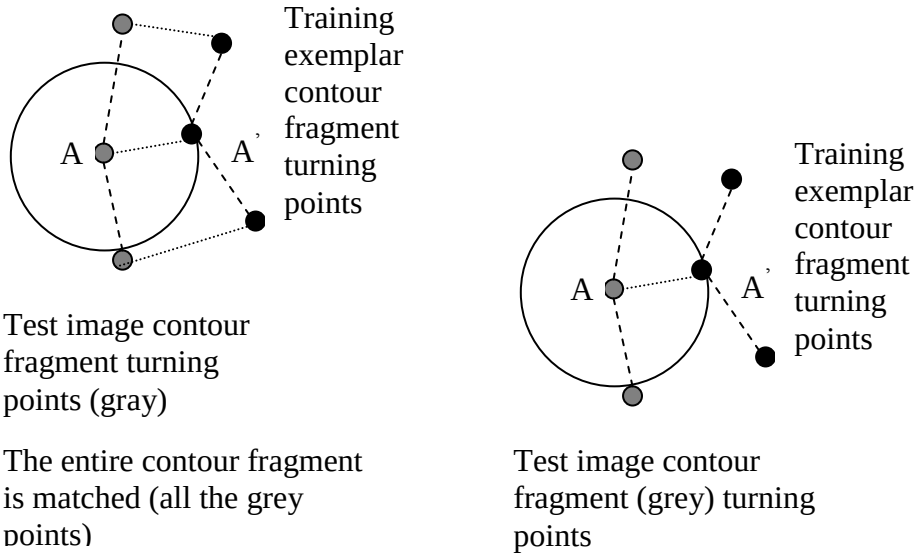


Figure 42. Turning Point A' is found within the neighbourhood of A;Fragment Matching: entire contour fragment is matched (left); Single Matching: only single turning point A is matched (right).

Feature matching is performed using sliding window of difference sizes across the image. Across all windows, the turning points at each location are matched with the merged exemplar turning points.

The turning points are matched in two ways (see Figure 42) which we call point matching and fragment matching. The first method matches the image turning point with the nearest turning point of the exemplars, within a small neighborhood window. For the second method, fragment matching, all the turning points in the curve fragment are matched to the exemplar contour fragment.

Given the following definition:

$T'(i)$	the i^{th} turning point from the sequence of turning points from the exemplar
$T(j)$	the j^{th} turning point from the sequence of turning points from the test image
L_S	location of the window of size S
r	the Euclidean radius distance from a turning point

The algorithm returns the location L, of window size S, where the matching is the maximum.

$$\underset{L_S}{\operatorname{argmax}} \sum_i \sum_j match(T'_{l_s}(i), T_{L_S}(j))$$

T_{LS} refers to the turning points at location L for window of size S. The exemplar, T', has a fixed window and size therefore its location and size is a constant of l_s.

The point-based matching is

$$match_S\ (T,T') = \begin{cases} 1 & D_{euc}(T,T') < R\ and\ D_{feat}(T,T') = 1 \\ 0 & otherwise \end{cases}$$

where D_{euc} is the Euclidean distance between the two points, and D_{feat} is the distance measure defined over the turning point features:

$$D_{feat}(T,T') = \begin{cases} 1 & T_\theta - T'_\theta < \alpha\ \ and\ D_{euc}\big(T'(i),T'(i-1)\big) - D_{euc}\big(T(i),T(i-1)\big) < \beta \\ & and\ D_{euc}\big(T'(i),T'(i+1)\big) - D_{euc}\big(T(i),T(i+1)\big) < \beta \\ 0 & otherwise \end{cases}$$

The parameters α and β are constants.

The fragment-based matching requires matching all the turning points from the same curve:

$$match_f(T,T') = \sum_{i+1}\sum_{j+1} D_{feat}(frag(T,i), frag(T',j)) + \sum_{i-1}\sum_{j-1} D_{feat}(frag(T,i), frag(T',j))$$

The function *frag(T,i)* returns all the turning points of the curve fragment that *T* belongs to. The current turning point *T*, is indexed as *i* and the previous points are *{0, .., i-1}* and the next following points are *{i+i, ..., N}*, when *N* is the number of turning points of the curve fragment.

The window location (Figure 45) with the largest match count is the probable location of the target object. Attributes calculated from this window are forwarded to a classification algorithm to determine if the target object is in the scene image. We also calculate the center of gravity of the all the turning points within the window. In addition, we store the area of the matched window, the average angular error of the matched turning points. For

the fragment matching method, we include the Euclidean length of the matched fragment, and the count of all matching the turning points of the matched fragment. All these additional attributes are forwarded to the classification algorithm.

4.5 Recognition Results

We use the Weizmann Horse database since segmented contour outlines are available. For testing against other categories we use the Broderbund ClickArt collection, and selected scene images of buildings and wildlife for testing.

We tested various classification algorithms using the open source Weka application. For most of the algorithms tested, the results do not vary much. So we present the best results found. We conducted tests using two slightly different turning points feature. The first, point matching, is just basically count to number of matching turning points from the exemplar around the neighbourhood. For the fragment matching method, once the matching turning point is found, we continue to find a match for all the turning points in that contour fragment.

	Point matching (365 instances: horse, buildings, wild)	Fragment matching (All 481 instances : horse, buildings, wild)	Fragment matching (348 instances: horse, wild)	Fragment matching (374 instances: horse, buildings)
AdaBoost with ADTree	84.9% (0.887)	97.5% (0.975)	97.1% (0.955)	97.3% (0.963)
Bagging with ADTree	87.4 (0.908)	97.9% (0.979)	97.4% (0.959)	97.3% (0.963)
MultiBoost with Decision stump	84.9% (0.889)	97.5% (0.975)	97.7% (0.964)	97.3% (0.963)
ADTree	N.A.	97.9 (0.979)	97.1% (0.971)	97.3% (0.963)

Table 4. Correct classification rate for horse and non-horse with the average F-measure in braces.

Single matching test has a total 365 instances consisting of 135 building, 107 wild, and 122 horse images. The fragment matching test uses 481 instances with 135 building, 109 wild and 239 horse images. The test was conducted using 10-fold cross validation.

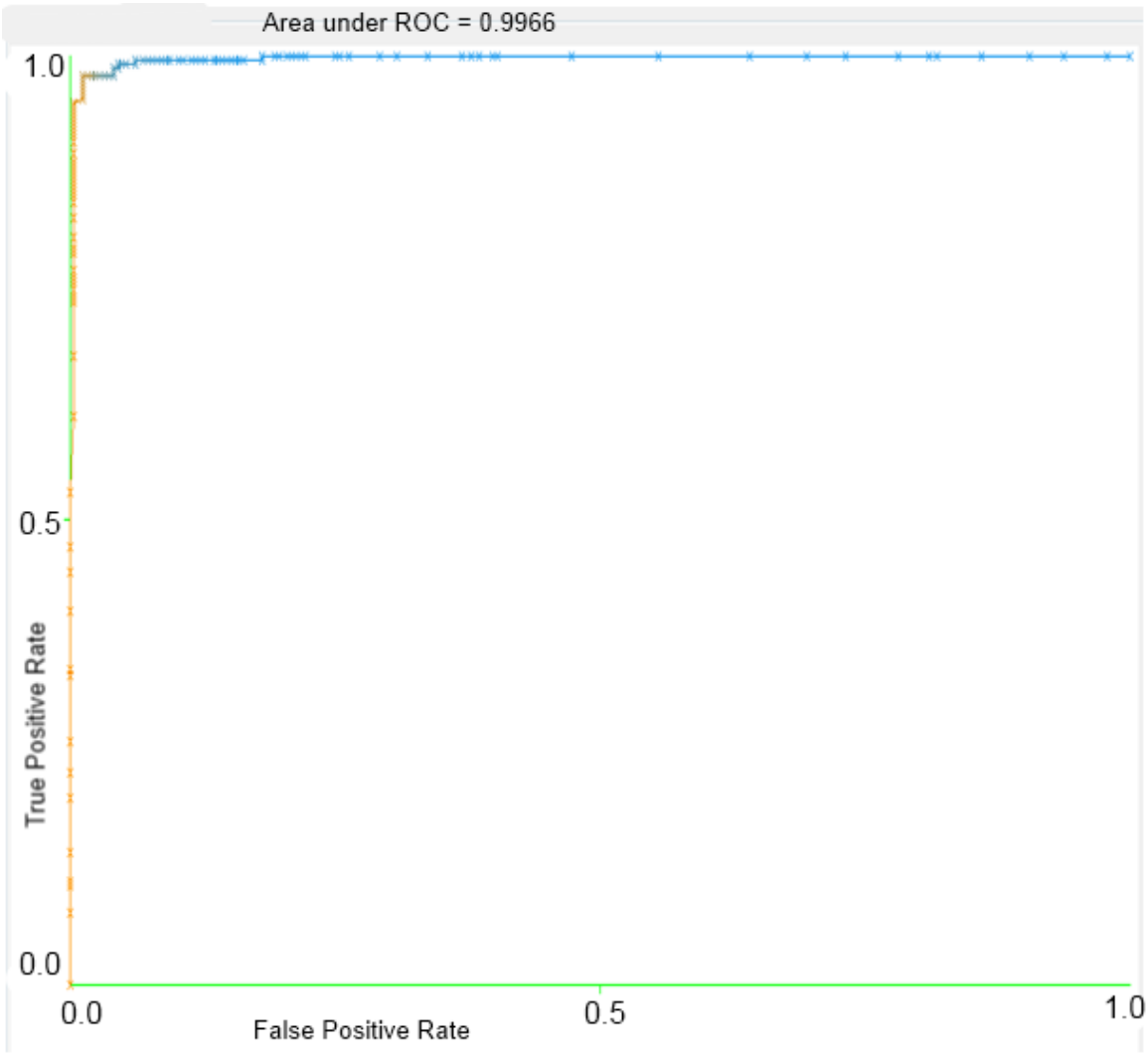


Figure 43. ROC curve for horse (ADTree classifier)

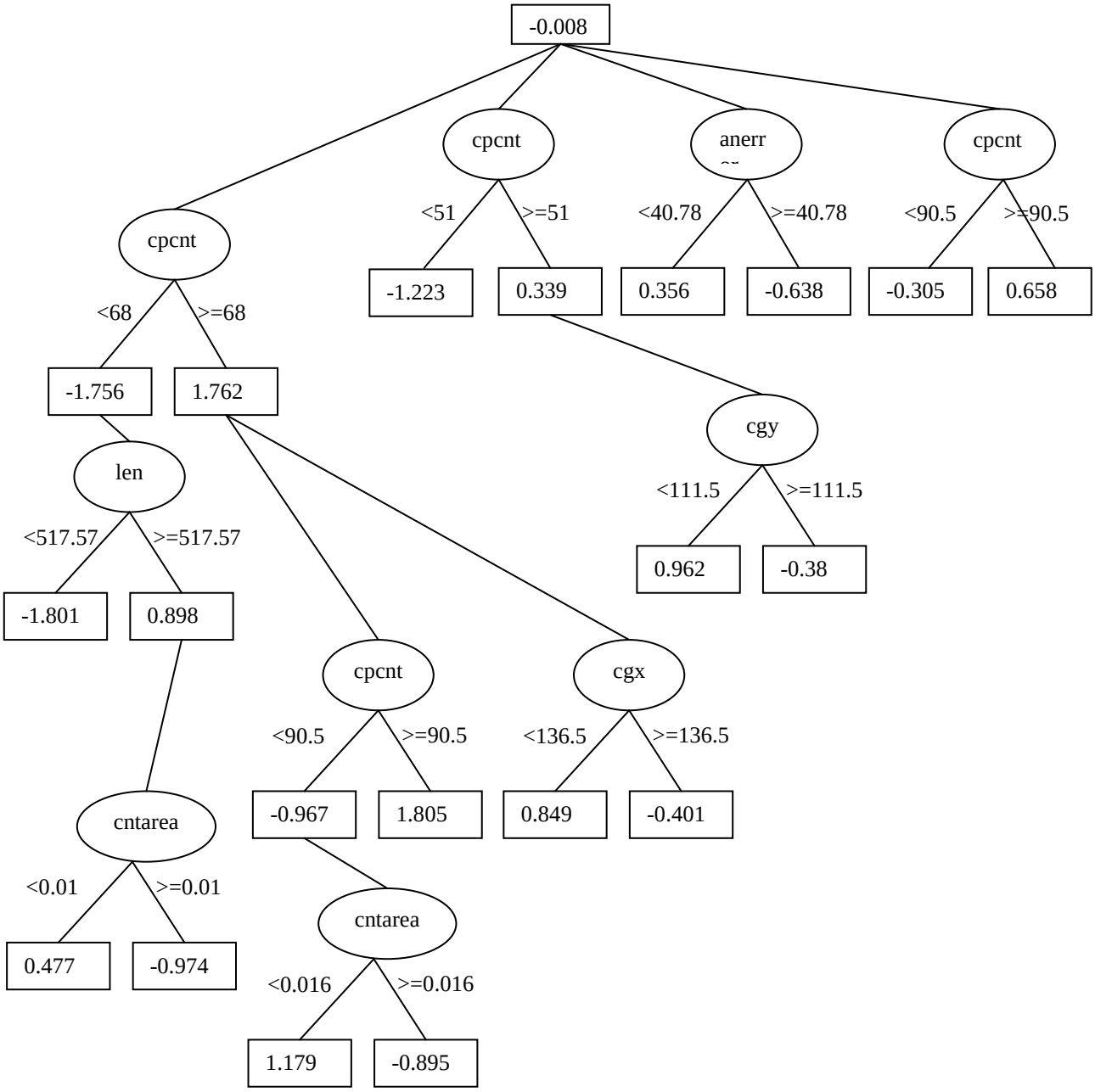


Figure 44. Classification tree generated

Figure 45. An image showing the best matched window results. The yellow points are the turning points of the test image, whereas the reds are the exemplar matching turning points, a short line links both of them.

We can evaluate the accuracy of the classification using the Receiver Operating Characteristics (ROC) curve and the Area Under ROC (AUC). The ROC (Figure 43) plots the true positives (y-axis) against false positives (x-axis) and is insensitive to changes in class distributions. A curve that bows more sharply towards the top left corner is a better classifier because it returns more true positives than false negatives. A classifier with diagonal curve is no better than random. Weka plots ROC curves by varying the threshold on the probability assigned to the positive class. The ROC gives a graphical depiction of classifier performance, whereas AUC gives a single scalar value that makes comparisons easier between classifiers. AUC measures the area under the ROC curve, and is equivalent to the probability that the classifier will rank a randomly chosen positive instance higher than a randomly chosen negative instance. A value of 0.5 indicates no better than random performance, whereas a score of 1.0 indicates perfect performance.

The distribution of the individual attribute values can provide insight on the quality of attributes (see Figure 46, Figure 47 and Figure 48). The graphs show a clear separation of values for horse and non-horse. Each of the two classes are unimodal with good separation between classes. The image that contains the target horse object is likely to have a higher count for matched turning points, and the length of matched fragment is likely to be longer. The counts and length is sufficient to account for most of correct identification of the target object.

The sample results in the next section shows the automatically generated bounding boxes in yellow. Note the count of matched turning points (red and yellow marks) in the horse and non-horse images. The count in the non-horse images is very much lower.

The classification tree in Fig 44 is generated from the alternating decision tree algorithm (ADTree) (Freund and Mason 1999; Pfahringer, Holmes et al. 2001). The attributes for this tree are total number of turning points matched (cpcnt), centroid (cgx, cgy), the angular error per point (anerr), average length per point (len) and the ratio of points per area of the bounding box (cntarea). The ADTree algorithm combines boosting and decision trees to create nodes of alternating decision (ovals in Fig 44) and prediction (rectangles). The overall score is combined from all prediction nodes to give a 2 class (position or negative) classification.

It is interesting to compare performance of this approach on zebras as they are basically horses with stripes. In the case where the full horse shape is clearly shown, the algorithm will correctly detect the zebra even though there are stripes. This is because the algorithm only counts correct matches and do not penalize incorrect matches. The results in Section 4.6 (th_anmwi077.JPG_box.png) show the detection of a zebra.

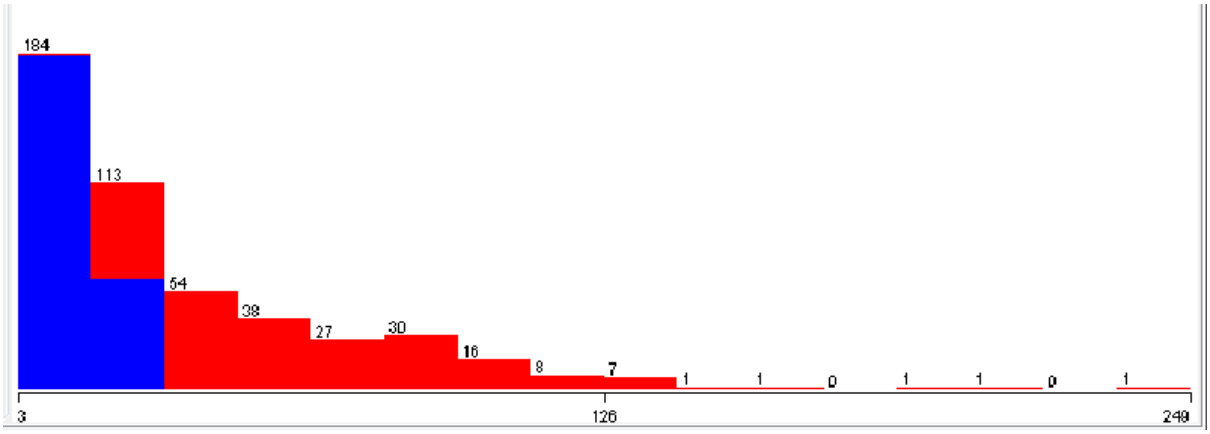


Figure 46. Histogram of the matching turning points count (X-axis), red is horse whereas blue is non-horse. Y-Axis: Number of images.

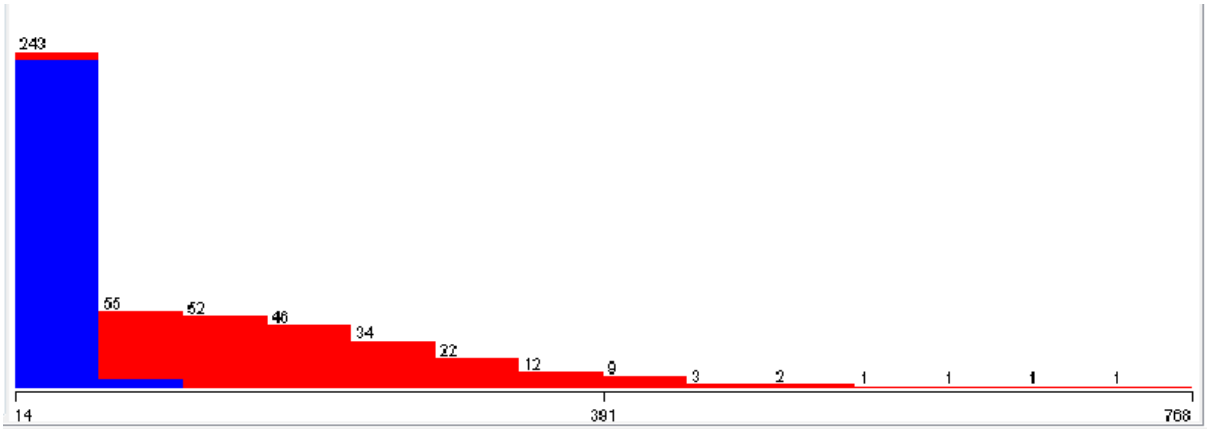


Figure 47. Histogram of values for matching turning points count (X-axis) within matched fragment, red is horse whereas blue is non-horse.Y-axis: No of images

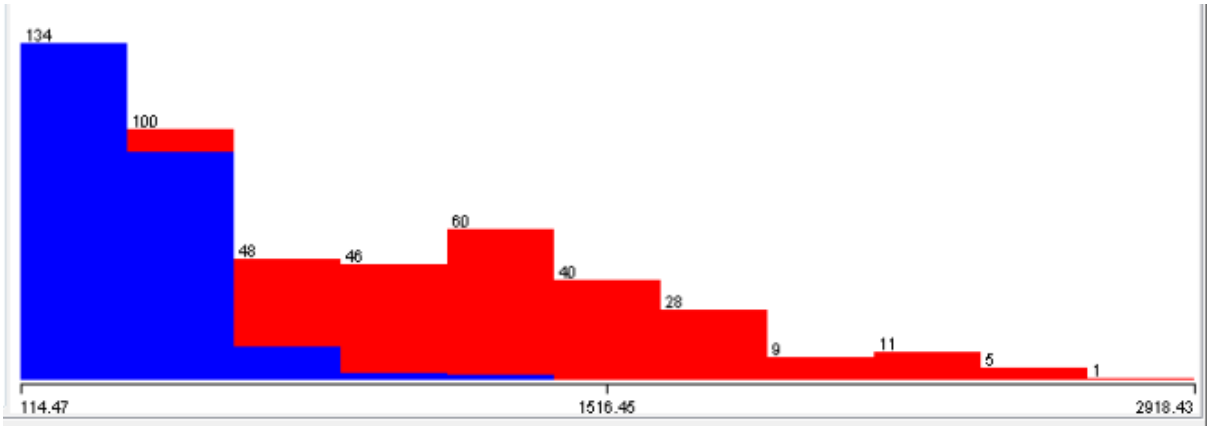


Figure 48. Histogram of values for the Euclidean length of matched fragments (X-axis), red is horse whereas blue is non-horse. Y-Axis: Number of images

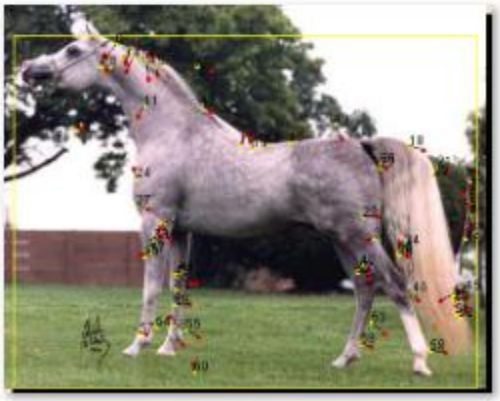
4.6 Sample Results

horse320.jpg_box.png

horse321.jpg_box.png

horse323.jpg_box.png

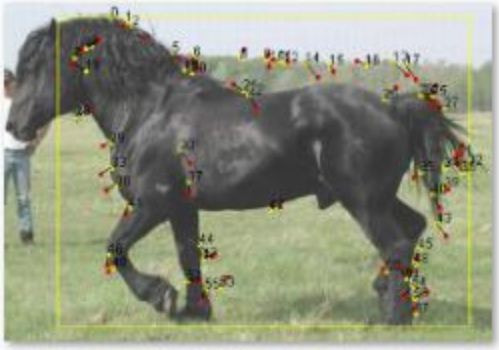
horse324.jpg_box.png



horse325.jpg_box.png

horse326.jpg_box.png

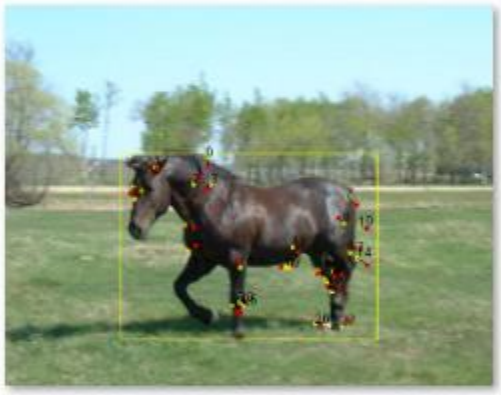
horse077.jpg_box.png



horse078.jpg_box.png

horse079.jpg_box.png

horse081.jpg_box.png



horse082.jpg_box.png

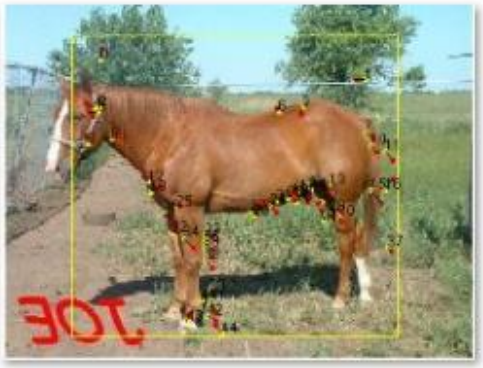
horse083.jpg_box.png

horse084.jpg_box.png

horse085.jpg_box.png

horse053.jpg_box.png

horse054.jpg_box.png

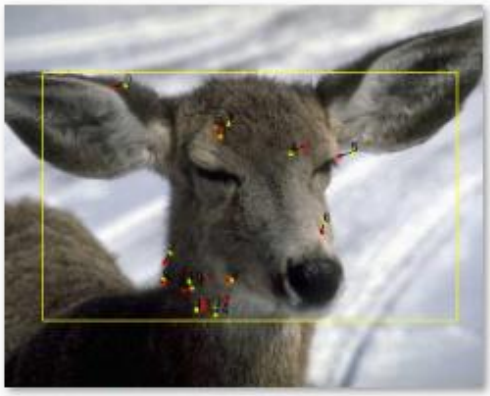


horse055.jpg_box.png

horse058.jpg_box.png

horse060.jpg_box.png

horse063.jpg_box.png

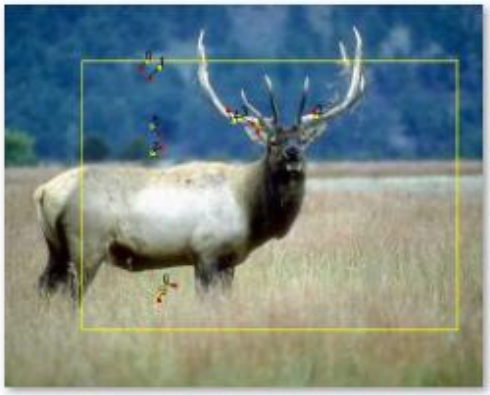


th_anmwi016.JPG_box.png

th_anmwi017.JPG_box.png

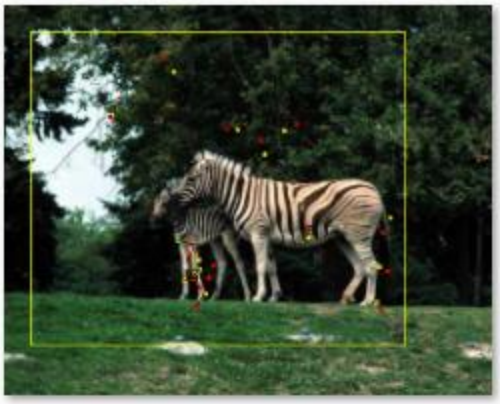
th_anmwi018.JPG_box.png

th_anmwi019.JPG_box.png



th_anmwi020.JPG_box.png

th_anmwi021.JPG_box.png

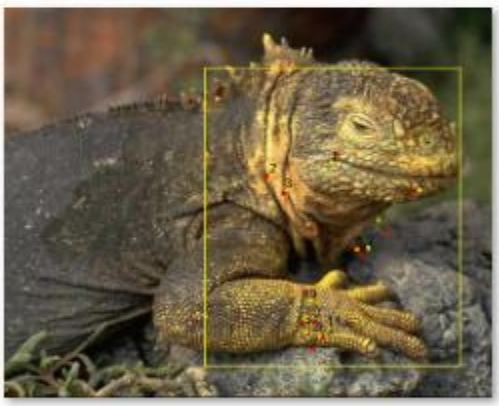


th_anmwi076.JPG_box.png

th_anmwi077.JPG_box.png

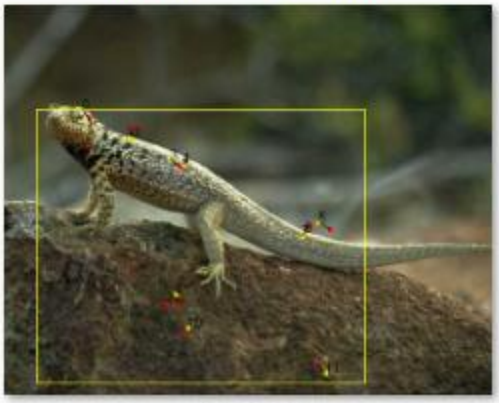
th_anmwi078.JPG_box.png

th_anmwi079.JPG_box.png

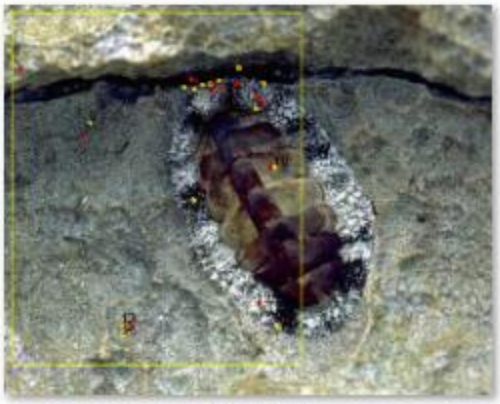


th_anmwi080.JPG_box.png

th_anmwi081.JPG_box.png



th_anmwi082.JPG_box.png



th_anmwi083.JPG_box.png

BLDHM033.JPG_box.png

BLDHM034.JPG_box.png

BLDHM035.JPG_box.png

BLDHM036.JPG_box.png

BLDHM037.JPG_box.png

BLDHM038.JPG_box.png

BLDHM039.JPG_box.png

BLDHM040.JPG_box.png

BLDHM109.JPG_box.png

BLDHM110.JPG_box.png

BLDHM111.JPG_box.png

BLDHM112.JPG_box.png

BLDHM113.JPG_box.png

BLDHM114.JPG_box.png

4.7 Background Texture Subtraction

Erroneous results can be obtained in cases where the background has sufficiently confusing edge line fragments (see Figure 49). One approach to fix this problem is to identify the background using image patches. We have previously worked on using discrete orthogonal polynomials and gray-level and RGB co-occurrence matrices to characterize textures, particularly textile textures (Cheong and Loke 2008; Cheong and Loke 2008; Loke 2009; Loke and Cheong 2009). We adapted it so that it can learn background textures. The image in Figure 50 on the right shows the result after morphological dilation and erosion of the background patches that have been recognized. The texture was trained from a different set of images.

Figure 49. Sample images showing erroneous placement of the object target window.

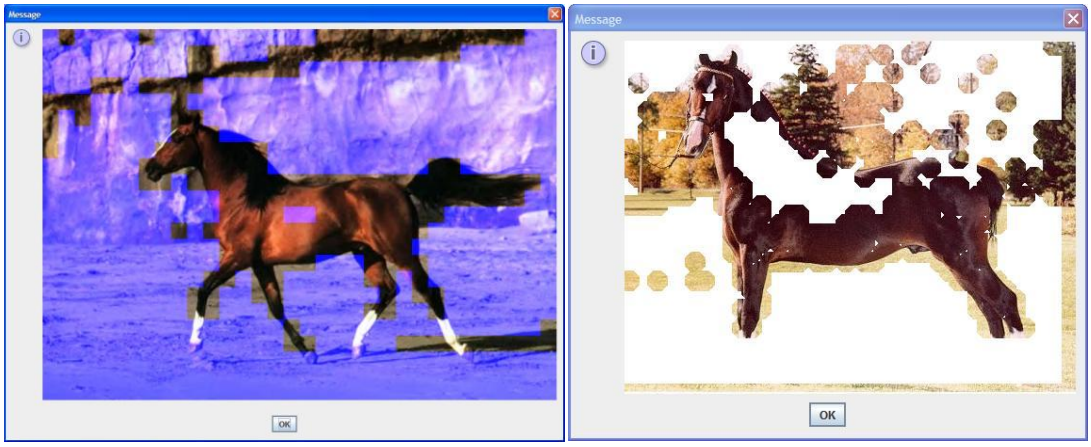


Figure 50. Results from Background texture identification. Left: The image on left with the background identified (purple). Right: Background (white) removed with a few morphological erosion and dilation operations.

The result obtained was not robust enough even though the procedure could identify most of the textures. The problem was that some of the identified background blocks straddle the object contours, and as such will remove those edges.

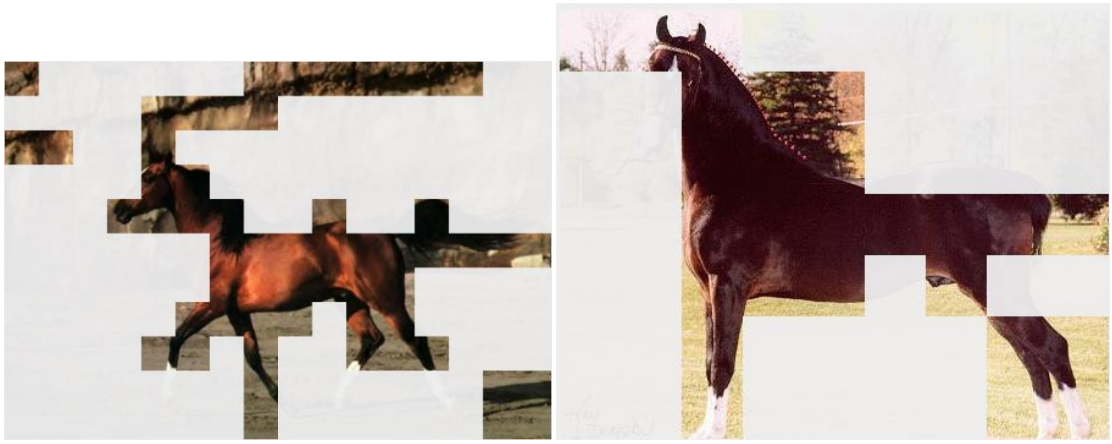


Figure 51. Background texture recognition using larger block sizes.

It is difficult to identify the block size (see Figure 51) and the number of morphological operations (erosion and dilation) required. Worse still the operation can remove the important contour edges. Therefore, rather than trying to remove the background entirely, it is better to just check if the turning points belong to the background or not. We tested this approach but didn't really improve on the recognition accuracy as the turning

points are good enough. However, this method may be useful for discarding non object edges when trying to reconstruct the object's contours (see next Chapter). However, we did not attempt this approach.

4.8 Discussion

There are other works that used turning angles for contour recognition, e.g. (Scassellati, Alexopoulos et al. 1994; Rusinol, dosch et al. 2007; Kpalma, Yang et al. 2008) but those works either tested on binary images or used turning angles which are based on technical or mathematical arguments, whereas our work are based on psychological and physiological research. We actually tested different versions of turning points but only presented the best version, and the version that gave the best results is consistent with psychological results of De Winter and Wageman (2008).

This work is closest with Shotton et al. (2008) work, and their results are so far the best, but compared to Shotton et al., our method is much simpler in terms of algorithm but with comparable results on the same Weizmann Horse database. Both approaches work well, despite the rather challenging images with background clutter; and wide variety of poses and sizes. The images that are misclassified are due to significant pose differences, the small size of the target object and similarity of the background edges to the training model edges.

Shotton et al use a total 228 horse images and Caltech 101 background set for tests, whereas we use 238 horse images against 244 animals and buildings images from the Broderbund 65,000 ClickArt collection. The Caltech 101 background category consist of assorted scenes around the Caltech campus and in the Vision lab, this should be comparable to the building images that we use because of the predominance of straight edges. The animal category that we use is likely to be more challenging because of similarity in backgrounds and curve lines lacking in campus scenes (in Caltech 101 dataset). Based on the published results (Table 5), our method achieved a better classification rate.

	Classification ROC AUC
Shotton-Canny	0.9127
Shotton-Berkeley Edge Detector	0.9275
Shotton-Boosted Edge Natural	0.9029
Shotton-Boosted Edge Horse	0.9518
Shotton (retrained)-Canny	0.9400
SVM-SIFT	0.8468
Our method-Canny-Fragment Matching	0.9966

Table 5. Comparison of classification results using Receiver Operating Characteristics (ROC) for Area Under ROC Curve (AUC)

The approach that we have is comparatively simpler than Shotton et al. even though there are similarities. Instead of building a codebook of contours we used turning points that made comparison easier as we are comparing points by points using the turning angle, Shotton et al. used a comparatively more complicated chamfer distance measure that required the contour need to be aligned, complicating the procedure. They used a boosted classifier, Gentle AdaBoost (Friedman, Hastie et al. 1998), whereas in our case a decision tree classifier produced just as good results without boosting.

We have shown that a simpler procedure produced as good results if not better when turning points of the contour fragments are used instead of using contour fragments directly.

We have also presented a theoretically justified edge boundary feature based on psychology and neurophysiological research.

Future Work

Even though the recognition method is fairly tolerant to slight pose variations; changes in the articulated parts can cause problems. One way to deal with this is to separate out the articulated parts, then that section will have a stack of image shapes corresponding to different articulation of the parts (see Figure 52). The proposed recognition process could compare the main body, and then the rest of parts which will consist of the many different articulation of the parts. This approach has similarities to the shape-tree where a tree is used to represent various hierarchical deformation of a shape (Felzenszwalb and Schwart 2007).

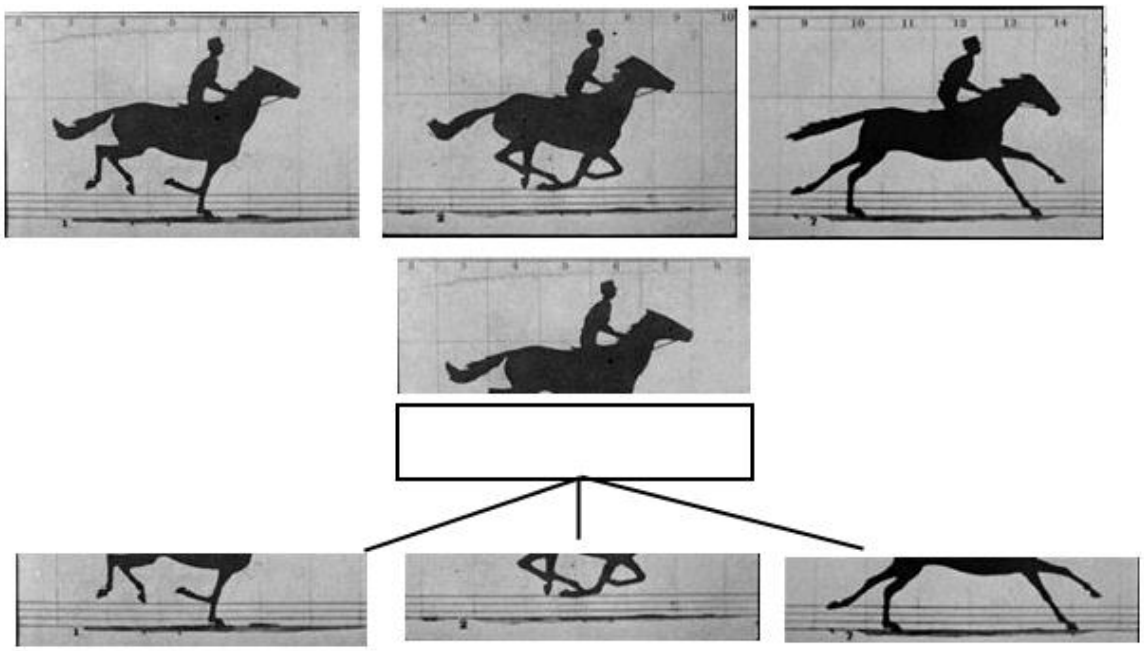


Figure 52. Dealing with articulated pose by separating the articulated parts. Image is Eadweard Muybridge's Horse in Motion from the Library of Congress Prints and Photographs Division. (This is image is now in public domain as copyright has expired).

The entire system can be parallelized to cope with different classes of objects, in different poses, and articulations of various scales. This is easy to accomplish because each sliding

window comparison is calculated independently. No special software configuration is required; each window calculation can be computed by computers connected in a cluster.

Chapter Five: **Object Contour Construction**

***Abstract* – The contour extraction based on identification of the object using shape-based features are presented. The earlier stage turning-angle based recognition returned a likely shape contour of the object. This information is used by an Ant Colony Optimization algorithm to return a possible shape contour of the identified object.**

Keywords- Object contour extraction, object contour detection, ant colony optimization, shape detection, contour detection

5.1 Introduction

The previous chapter developed a new algorithm for object recognition using intensity edges. We have shown that the approach using turning points for object recognition worked very well. The object segmentation will now be guided by (1) the edges as identified as part of the object contour, and (2) the exemplar shapes that was used in the recognition.

We now have a set of intensity edges, some of which we have identified to be part of the object contour. However, we do not definitively know which ones, only that a majority of them belong to the object. Some of the edges identified may be wrong, but the majority is correct (See Figure 53). We also know the probable shape of the object but not with accuracy. Given this we want to try to a contour that fits the scene given the constraints above. This can be seen as a constraint satisfaction problem but with the constraints themselves as uncertain. This seems like a problem that we can't solve easily without some kind of trial and error.

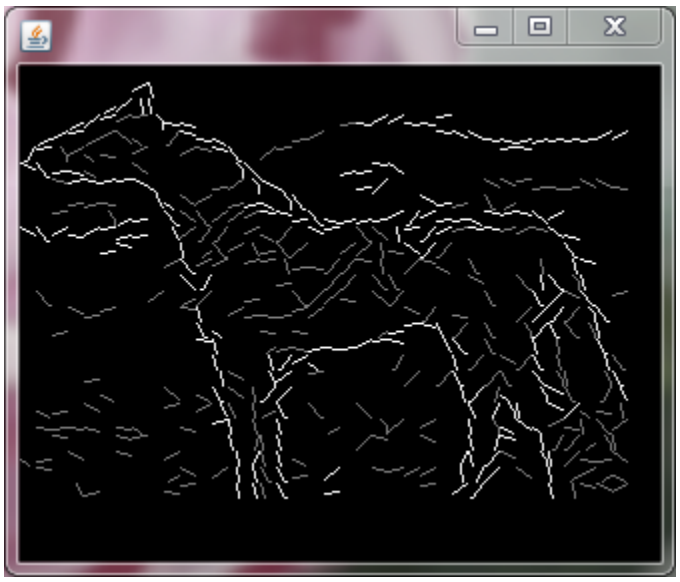


Figure 53. Wedgelet image with the edges; highlighted edges are the detected edges that are likely to be part of the object contour. Clearly some are misclassified as object contour.

For a solution to this problem, we attempt a solution using ACO. Given the large search space that is possible, we need to reduce the problem to a suitable size. The image is converted to a 8x8 wedgelet representation that was discussed earlier (Chapter 2), this basically reduces the image to 1/8 in width and height. The edges detected are marked in the corresponding wedgelet image.

5.2 Implementation

5.2.1 Ant Net initialization

The entire wedgelet image is mapped to the ant network. Instead of initializing the pheromone randomly, they are seeded according to the edge strength. The edges not

marked in the recognition process have their initial pheromone strength reduced (we used 0.4 of the edge difference).

5.2.2 Path construction

The ant in the ACO algorithm needs to construct a path that represents the object contour. Apart from the constraints given in the ACO algorithm we want to impose a few general constraints when constructing the path. The reason is that there are too many degrees of freedom available that will create undesirable paths. The constraints we use are:

1. The path should be as smooth as possible. The reason this is desired is to prevent convoluted paths that folds back and forth, and natural objects generally have smooth curves.
2. The path should not crossover or turn back on itself. This is to prevent creating a space-filling path with no internal spaces.
3. The start and end node should be close to each other. This is ensure a closed loop contour

To satisfy constraint (1) we simply require that the constructed path follows in the same general direction as much as possible. For example, if the path was previously heading from right to left, then it should be constrained to only three possible choices on the left (the shaded squares in Figure 54) out of the 8 possible neighbouring points.

Figure 54. Left, path choice (shaded) for a horizontal directed path; right, path choice (shaded) for a diagonal directed path.

When the directed choice path points are blocked, where the path has reached the border, or crossed back to itself, then only other path choices will be made available. For instance, in the above example, if the shaded points are not available, then the non-shaded points will be considered.

We also need to check that the path does not cross back on itself, and that it does not get too close to itself. This is accomplished by checking that the possible neighbouring points distance exceeds a certain distance threshold from all its previous path points. Points that are too close to existing path points are not permitted.

These constrained available choices are then passed on to the ACO algorithm to pursue based on its optimization scheme of exploitation or biased exploration. The desirability of a path point (or node) is based on the original edge strength and deposited pheromones. Finally the path is terminated if the next point is close to the starting point.

5.2.3 Path quality evaluation

After the path is constructed the quality of path is evaluated. We want to evaluate the quality of the path by its similarity to the shape of the exemplar. The exemplar is the model shape that will be used to guide the object segmentation. Basically the path should be converging to the exemplar while being constrained by the actual edges. This implies that the path quality measure should be some shape fidelity measure.

For this measure we perform chamfer matching using the distance transform. Chamfer matching is a popular technique used to find alignment between to edge map. In our case we use one edge map from the exemplar, and use that to match with the ant-constructed path. Chamfer matching provides a continuous measure that can tolerate small misalignments, occlusions and deformations. The chamfer distance is efficiently computed via the distance transform. The distance transforms (DT) calculates for each point x, the distance to the nearest edge, x_E:

$$DT_E(x) = \min_{x_E \in E} \|x - x_E\|$$

The chamfer distance gives the average distance from the points, x_p in the ant-constructed path to the closest exemplar edge. The chamfer distance is then simply given by:

$$D_{chamfer}(x) = \frac{1}{|x_P|} \sum_{x_p} DT_E\,(x_p)$$

This is simply a lookup operation given the pre-calculated distance transform from the exemplar. This distance (Figure 55) has to been calculated each time an ant constructs a path, so it needs to be efficient.

Figure 55. Distance transform of the exemplar with the ant constructed path (white) superimposed. The darker squares show nearer distance to the exemplar contour.

However we need to compare it in a scale-invariant manner, since the target object can be of different sizes. Therefore, before calculating the chamfer distance, we rescale the size of the path to the size of the exemplar template (see Figure 56).

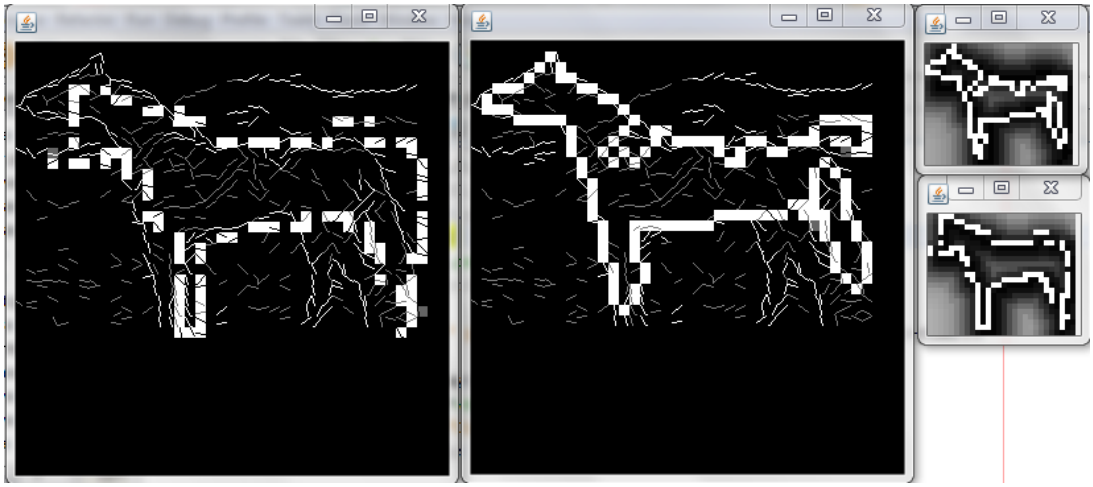


Figure 56. (Left) Path of the exemplar superimposed on the test image. (Middle) The constructed ant path over the wedgelet image. (Right) Paths superimposed on the distance transform of the exemplar.

5.2.4 Ant Colony Optimization (ACO) algorithm

For the ACO algorithm we use 20-40 iterations per epoch, of 1500-3000 epochs. At the end of each epoch, the best ant is allowed to update the pheromones on its path. If this is the best all time ant, then it updates the full amount, else a fraction of full amount is applied (we use 0.5-0.7 of the full amount). The update amount is the scaled value from the chamfer distance value. At each epoch all nodes in the ant network are evaporated.

5.3 Results and Discussion

Figure 56 shows the comparison of paths constructed by the Ant algorithm (middle) against the exemplar path (left image). Both of these paths are superimposed on the test image wedgelet pattern. It can be seen that the Ant-constructed path differ rather significantly from the exemplar path (compare the 'head' section). This shows that the algorithm can generate flexible solutions according to the constraints. Also the test image has rather significant edges misclassified (see Figure 53) in the prior recognition process, note especially that some of the background 'hill' has been classified as part of the horse. This misclassification would not have happened if we have done the comparison process

separately for each shape exemplar instead of comparing it together (see Figure 37; where all the exemplar turning points are fused together). This caused the detected contour to have multiple edges for the 'back' of the horse because the exemplars are a fusion of all the exemplars with multiple outlines. In any case, the Ant algorithm is able to reconcile it and obtain the object contours.

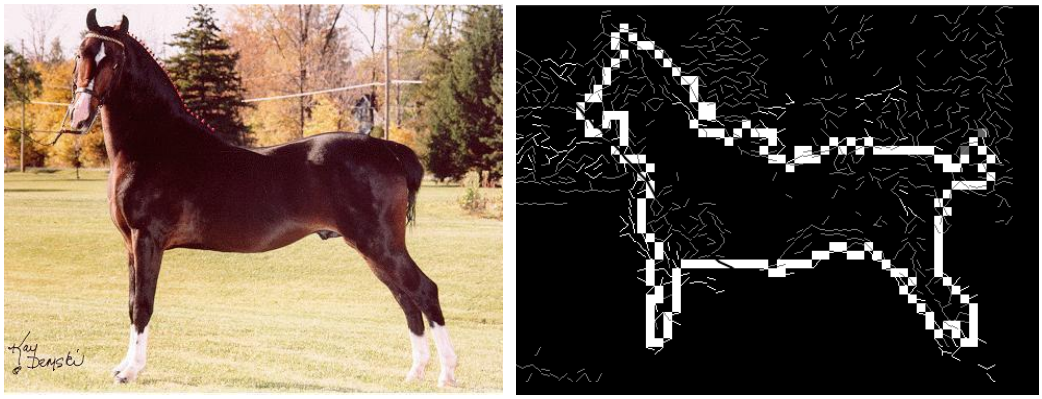


Figure 57. Original Horse004, (right) Contour detected for Horse004

Figure 58. Actual size of Horse004 image with the ant-constructed contour superimposed

Figure 59. Actual size of Horse027 with the ant-constructed contour superimposed.

Figure 60. Actual size of Horse022 with the ant-constructed contour superimposed. (Right) The exemplar used by the ant to construct the contour.

The results provided are a sampling of contours that we have extracted. Our purpose here is to demonstrate our concept of our approach, and not to evaluate the viability of the ACO algorithm. It may be that there are better approaches that could be used to

determine the object contours given the constraints. In the future, we will explore other optimization techniques.

The results show that it is possible to obtain contours in a scale invariant manner (see Figure 57, Figure 58, Figure 59, Figure 60 consisting of images of different sizes). Though the results are not perfect, it is possible with some additional post processing, such as smoothing and constructing a convex path, some defects can be ameliorated. Typical number of paths constructed and examined was in order of about 10,000 paths.

There are some weaknesses in the ACO approach. For example for larger images, the search space is too large for the ACO, and it becomes sensitive to initial conditions. The path contour quality evaluation is critical in determining its success. The chamfer distance was calculated using the average – this means that some path that could give the same results even though the path differs, this is as long as the further path points gets cancelled out, returning the same average. The other error is due to the rescaling error. The exemplar that we use is smaller than the image being tested.

The results could be improved upon with better path measures. However, we did not have resources to test other evaluation functions. The results show that it is possible to recover contour paths, and that is sufficient to make our case. In order to deal with shapes with greater variability, multiple exemplars could be used.

For comparison, we show some of results by other researchers (Figure 61, Figure 62). The detected contours from these results show that they do not vary much from the training examples. Our results have a wider range of tolerance to contour variations for the detected contours. In Figure 60 (right), the exemplar (training model) is shown, the detected contours in Figure 58, Figure 59, Figure 60 differ rather substantially.

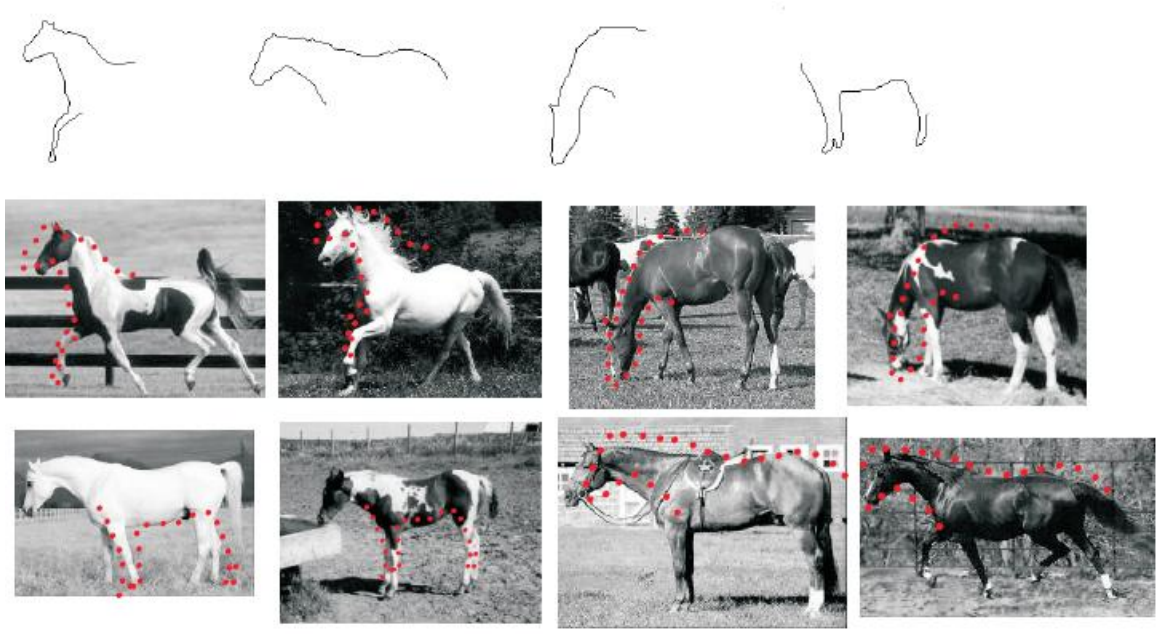


Figure 61. Contour fragment detection. (Top) Contour parts used for detection. (Bottom) Detection results using the Shape Band method (Bai, Li et al. 2009).

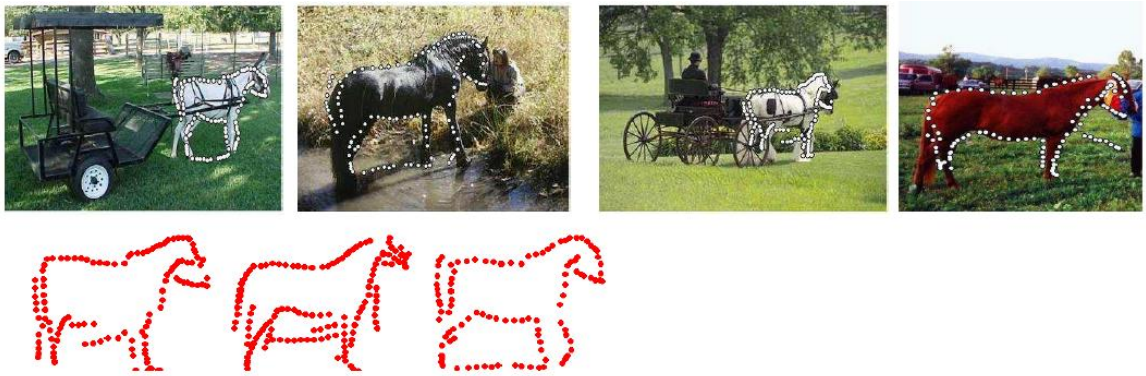


Figure 62. Contour detection using Pairs of Adjacent Segments (Ferrari et al 2010). (Top) Results of detection. (Bottom) Models used for detection.

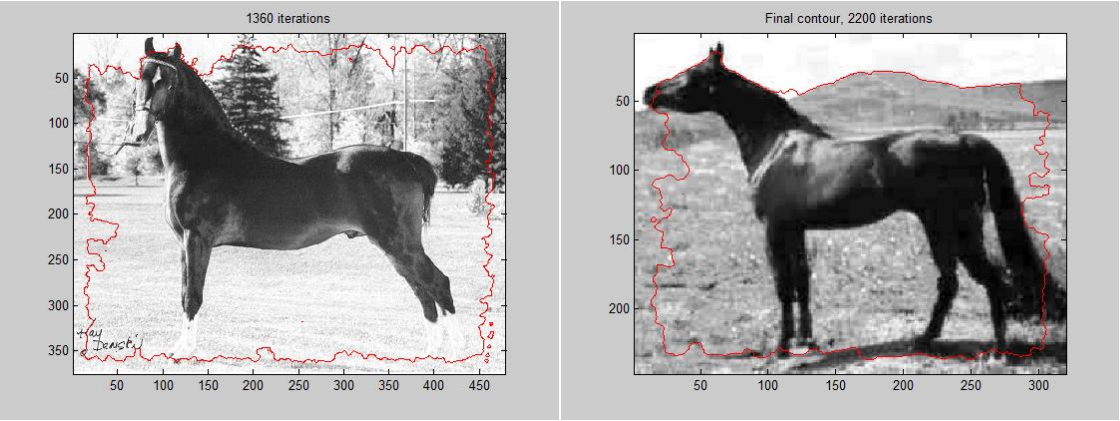


Figure 63. Contour results from curve evolution using level set.

Figure 63 shows the result from using the level set algorithm ((Li, Xu et al. 2005). The number of iterations was chosen so that the curve remained fairly stable and do not change much. The results show that the basic curve evolution algorithm is not able to detect object contours unless the starting contours are placed almost outlining the object (as in Figure 29 bottom-left).

Chapter Six: **Top-down bottom-up Image Segmentation**

***Abstract* – Review of psychological and neurophysiological research on top-down interaction recognition is presented. An implementation framework based on our work is discussed.**

Keywords – feedforward models, top-down interactive models, image segmentation

6.1 Introduction

The core question we investigated was to find out how to perform better image segmentation of natural scene images. In order to limit the scope we opted to look at images with a single dominant image but in a natural setting as opposed to in a contrived environment whereby, say, the background can be easily segmented because of artificial constraints. Related to the question of how is the question of what? In particular, what other information would benefit image segmentation. Obviously this is a huge question, and we can only explore a path towards it. We would like to summarize our journey and the particular reasons why we took this ride, and the valid lessons that we learned along the way.

In order to understand better image segmentation we need to understand what is segmentation and why it is difficult. Firstly we also have to ask what the goal of image segmentation is. Very simply it is the delineation of different objects in a scene image. The goal is to aid automatic understanding of the scene image by bridging the so-called semantic gap, i.e. to provide meaning to a bunch of organized coloured pixels.

Two common approaches are used for image segmentation – region-based, or intensity-based. The region-based approach seeks to amalgamate regions of similarity together as a single unit. However this method will fail to segment out the object unless the object has only a single contiguous region that is similar. Otherwise, several regions need to be

bound together. Then the question is then: how is this to be decided? The same question will be asked for the other intensity-based approach as well. Since we are not taking the region-based approach, we will not discuss this further.

The other approach is to work with intensity differences. Intensity differences along an axis give rise to edges. This creates a problem because edges as intensity differences do not always conform to object contours. Some of these edges are just incidental edges arising from lighting particularities and from texture, others are object internal edges. How we can distinguish them from actual object contour edges? A couple of approaches have been attempted:

1. Get external help. This approach requires manual intervention by performing a rough segmentation.
2. Train them. The segmentation is aided by past successful learned segmentation.
3. Use some properties (internal, external or global). Usually this is the Gestalt properties, like good continuation, etc.

The first approach is not useful for task automation. The second attempts to learn what a true edge is in general based on what it have learnt, but will not identify the object contour. The third approach is the approach we taking.

6.2 Feed-forward Models

We follow a rather well-threaded path of using edge-linking with a little twist of our own. Firstly we used a wedgelet representation which is a geometrical approximation of the actual image. The image is subdivided into squares, and within the squares we used almost triangular-like wedges to approximate the content within the squares. The wedge lines are used as a stand-in for edges. This approach has a few advantages. One, the noise element is reduced because within the square block, it is guaranteed that there is only one wedge line. Second, we have reduced the representation, instead of working at pixel level, we are effectively working on the block level, and this reduces computation when

searching for edges to link. Third, we have introduced a texture component, so some of texture will not be confused with an edge. Now comes the edge-linking part where we try to link the edges into meaningful contour parts using a non-deterministic algorithm called the Ant Colony Optimization. This approach still follows the standard paradigm, i.e. from edge to segmentation to objects. The assumption is that edges will eventually lead to object segmentation if we can link them by some rule. The ants trying to link up the edges make the decisions on lower level properties such as intensity differences, path continuity, and general shape parameters like roundness. It seems to us that the constraints are under-specified, and that they cannot work in all cases. For example, it is not the case that the strongest edge will or should be a part of an object contour. How do we find the cues and constraints that will guide the edge linking?

David Marr (1982) proposed that a three level framework that underpins visual perception. He posits that visual perception can be analyzed and described in these levels: computational, algorithmic and implementation. Computational relates to the high level task that the visual system is solving, while the algorithmic relates to the representation and its manipulation required to solve the higher level computational tasks. The implementation level refers to the physical part of the system.

He also proposed that the retina image undergoes various processes namely; the primal sketch, 2.5D sketch and 3D model representation, in increasing sophistication. Basically it proceeds from the primitive representation of edges, boundaries and regions in the primal sketch to the 2.5D sketch which now factors in orientation and depth of visible surfaces. The final representation is finally accomplished by integrating 3D views and models.

Another influential model is Biederman's (Biederman 1987) recognition-by-components (RBC) approach. It also proceeds in a bottoms-up manner, similar to the primal sketch, using simple and easy to compute information such as edges from surface characteristics that leads to a line description of the image. The assumption is that object edges are

stable across variation in the image, and from these, stable invariant 3D models could be derived. The theory suggests that the processing path is feed-forward. Serre and colleagues (Serre, Wolf et al. 2007) have implemented a biologically accurate feed-forward model made up of simple and complex cells that predicts human performance.

The fact that recognition takes place within 150-300ms (Thorpe, Fize et al. 1996), led many to suggest that only feed-forward mechanism are involved as the short-time frame imposes serious constraints on possible feedback mechanism. Therefore a feed-forward only approach seems the likely candidate. However, in ambiguous cases, e.g. clutter, foreshortened views, or poorly illuminated, might require prior assumptions to be available. Some researchers postulate that implicit priors are used all the time and top-down priors some of the time, depending on task demands (DiCarlo and Cox 2007).

6.3 Top-down Interactive Models

Are there enough local and signal-based information in the edge-extraction process that allows for object recognition? Sanocki (Wang, Bowyer et al. 1998), through their experiments, answered it in the negative. In fact, they wrote: "The results call into question psychological and computer vision models that use local edge extractors as their only low-level process". Graham Davies and his colleagues have reported (Davies, Ellis et al. 1978) that images which contain exclusively contour information are very difficult to recognize. They found that subjects could recognize only 47% of the line-drawings compared to 90% of the original photographs. These results have been replicated in Jian Wang et al. (Wang, Bowyer et al. 1998) and Sanocki (Sanocki, Bowyer et al. 1998). There is evidence that humans when asked to draw contours, the contours often include depictions that contain significant photometric cues. Bruce and colleagues (Bruce, Hanna et al. 1992) show that the contours drawn by accomplished artists include not just to a low-level edge-map, but also include the photometric structure that is not obtained by edge map.

It is known that object recognition is invariant under a variety of conditions. The presence of shadows have little effect on recognition rate (Braje, Kersten et al. 1998). This suggests a consistent representation that is immune to shadows, and argues against an image-based system that performs shadow detection and labeling. This would suggest that using only edges could be ruled out because of the need to differentiate edges caused by the object and shadows. They also rule out colour, texture and boundary sharpness as features used for representations. They suggest that a global cue that is not easily disrupted locally by shadows is used, such as global shape.

There are numerous studies that reported that human visual system treats upright faces differently (Yin 1969; Diamond and Carey 1986). There is evidence that face recognition is processed in a specific area in the visual cortex (Tsao, Freiwald et al. 2006). The fact that we are capable of recognizing upright faces more easily than a vertically inverted face argues for the case that face recognition is not simply feature based since in both cases the features remain the same. It is argued that face recognition is configural or "holistic" in that both parts and relationship cues are integrated into a single recognition task (Young, Hellawell et al. 1987; Tanaka and Farah 1993).

There are various psychophysical experiments that support top-down facilitation for image segmentation and object recognition. Peterson (Peterson 1994; Peterson and Gibson 1994) suggests that object recognition process – called prefigural recognition - occur before figure-ground segregation. In figure-ground analysis, the objective is to separate out the object from the background, before the object is recognized. This is generally assumed in the theories of Marr (Marr 1982) and Biederman (Biederman 1987). Peterson and colleagues propose that edges are detected early so that object recognition processes can be carried out in parallel with figure-ground analysis. This idea is counter to leading computer theorists like Marr and Biederman. They assumed that figure-ground organization must precede recognition.

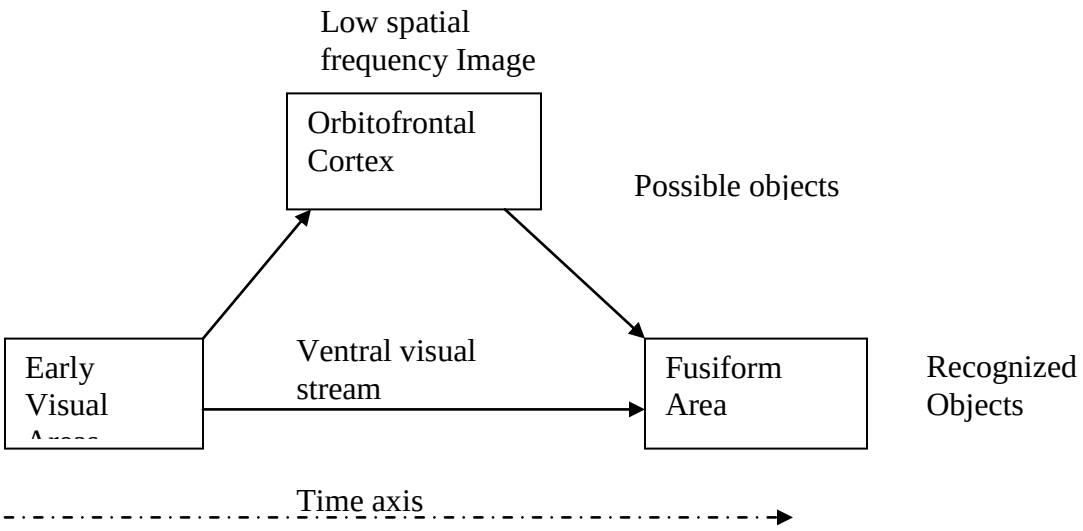


Figure 64. An illustration of the role of the orbitofrontal cortex in object recognition (based on Bar et al 2006).

Vecera and Farah (Vecera and Farah 1997) report that image segmentation can be influenced by the familiarity of the shape being segmented. They also show that low level grouping suggested by connectivity and common region can be overridden by familiarity. Their results are consistent with the hypothesis that image segmentation is an interactive process. Vecera and O'Reilly (Vecera and O'Reilly 2000) gave an account of the figure-ground organization using interactive interaction (Rumelhart and McClelland 1982; Rumelhart and McClelland 1986). They argued that both top-down and bottom-up forces are activated simultaneously so that object recognition need not fully occur prior to figure-ground segregation. Graboi and Lisman (Graboi and Lisman 2003) also developed an interactive interaction model for visual recognition of words using top-down and bottom-up processes. However their model did not use real-world images but artificially constructed letters made of 8 segments.

Neurological mechanisms (Figure 64) have been suggested by Bar et al. (M.Bar, Kassma et al. 2006) that involve top-down processing. They proposed that the orbitofrontal cortext (OFC) is used in top-down facilitation of object recognition. The OFC is activated

by a lower resolution image via the magnocellular pathway that connects to the early visual processing areas. This coarse representation is used to activate the most likely interpretation of the input. This "initial guess" combines with bottom-up (low level) cues to complete the recognition process. The hypothesis is that the OFC serves as a rapid detector and predictor of potential content based on coarse aspects of the input. Other evidence that supports the OFC facilitation of early recognition are found in the review by Fenske et al. (Fenske, Aminoff et al. 2006).

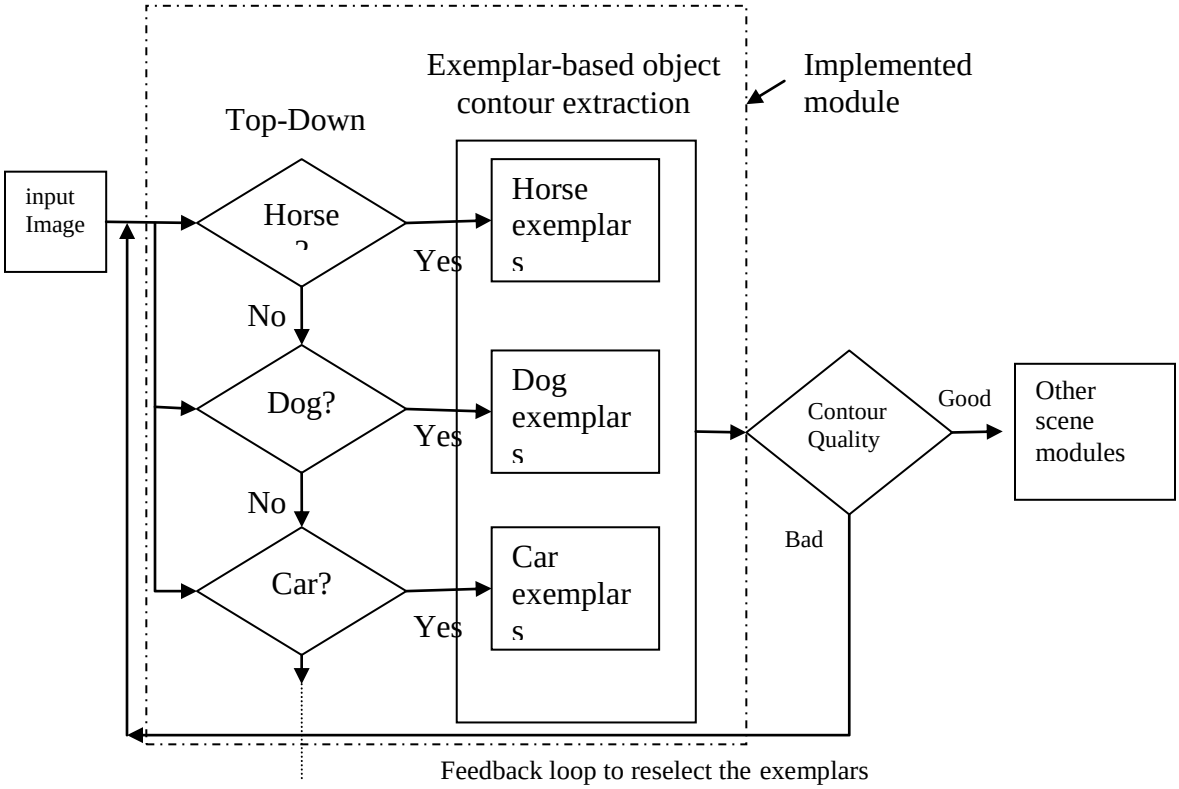


Figure 65. Block diagram of a top-down bottom-up segmentation by object contours.

Therefore it seems that the human visual system co-opt various mechanisms for object recognition, including top-down facilitation. Whether human visual object recognition is only feed-forward or if it includes top-down interaction is currently under vigorous

debate. However our purpose is not to construct a neurophysiological account of the human visual system, but simply to gain insights from biology. It stands to reason that if we want computer vision systems to be on par with biological systems, we need to learn more from it. Based on the research, the use of top-down facilitation with bottom-up cues for segmentation should be subject to further research.

6.4 Top-down bottom-up Implemention

We have implemented the extraction of contours that can be facilitated by top down knowledge in a computational program. We have shown that this approach is viable by proof of concept. However, our approach is not meant to be biologically accurate. Our approach, in general, works this way (see Figure 65): Given a scene image, we attempt to identify the possible object in the image, then pick the most likely object and use the object's shape as exemplar for contour extraction. The quality of the contour is measure, if it is good, then the task is complete, else try another shape exemplar. We have not implemented the feedback model, but we have demonstrated that top-down bottom-up contour extraction.

The object recognition phase works by detecting edges using standard intensity-based edge detection technique (i.e. Canny edge detection). From the edges detected we discard the shorter edges. The longer edges that are kept are analyzed for turning points. We have discussed the literature that demonstrated turning points as biologically salient points. We used these biologically salient turning points for preliminary scale-invariant object recognition or detection. The comparison stage uses a shifting window to compare turning points from a set of reference image turning points. This process is fast and can be parallelized (see Figure 66) because each shape and window can be compared independently.

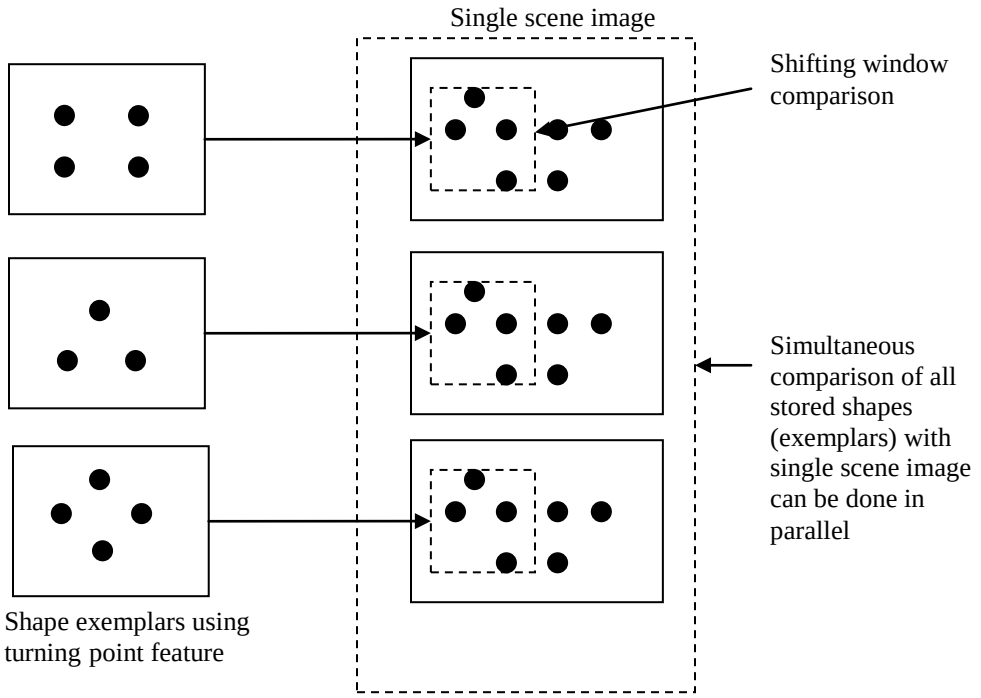


Figure 66. Parallelizing comparison in object recognition.

The comparison process that we used is similar to the sliding window approach used in the Ada-boosted real-time face recognition (Viola and Jones 2004), yet simpler; instead of comparing image patches we are simply comparing and counting vectors of points. Since the mechanism is similar, we can benefit by adopting real-time Ada-boosted learning and recognition mechanism. This could be done in the future. We have shown that our approach is scale-invariant and is robust against deformation that is not too violent. However it is not pose invariant; pose and parts articulation can handled as discuss in the earlier chapter, basically by additional comparisons that could be handled in parallel.

Chapter Seven: **Conclusion**

We now revisit the research questions that were raised initially. To recap we asked:

How can a model of automatic object contour detection and extraction from scene images that include contextual and possibly prior information improve significantly an object contour-based image retrieval system?

First the subsidiary questions:

1. What type of contextual information or prior information that is appropriate?

   We did not fully explore all the various contextual information, as that is clearly not possible, but based on preliminary experience we discarded the use of local contextual cues. We worked using prior information, i.e. the possible identity of the object to be segmented. It is also possible that other contextual cues could be included to provide clues to identity of the object, but this was not explored.

2. What is the interaction between object detection and contour extraction given that both are inter-dependent?

   The identity of the object will inform the contour extraction algorithm of the possible contour, and in turn will guide the segmentation and extraction of the contour. The quality of contour (by comparing with training exemplars) extracted and this could be used as a means to accept or reject the contour. If it is rejected the next possible object would be used to guide the segmentation.

3. What is an appropriate representation and properties of the object contour?

We have found, based on neurophysiological and psychological evidence that the angle of the turning points measured between salient points provided good representation of the contour, as evidenced by the recognition results we obtained.

7.1 Contributions

Putting all this together (see Figure 67), we propose a model that has a recognition phase that precedes actual object contour segmentation. This model is based on experimental and theoretical considerations. Psychological and neurophysiological research suggests that recognition may occur interactively guided by top down processes from bottom up cues.

Our recognition module uses turning points of curve fragments for shape-based recognition that we developed. The recognition module provides the top-down influence for segmentation. Our model performs segmentation by combining top-down influence and bottom-up cues as suggested by psychological research. We used a new image representation system using wedgelets coupled with an Ant Colony Optimization algorithm for contour extraction. The object contour is formed from edges detected and based on local continuity cues guided by the exemplars provided by the recognition module.

The framework allows for self-quality checks. The result of the segmentation can be evaluated to check that the contour extracted conforms to stored models. The contour model also interacts with the recognition model by providing information about the quality of the contour. The next possible model could be activated by the recognition model if the contour extracted is not satisfactory.

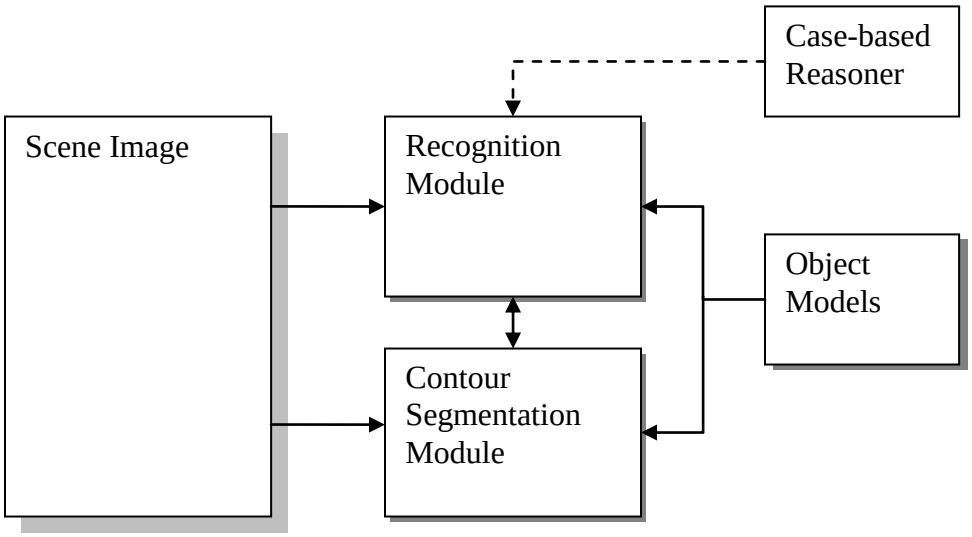


Figure 67. Model of a Recognition-based Object Contour Segmentation with Top-Down and Bottom-Up processes.

7.2 Future Work

The recognition module could optionally include patch-based (e.g. using SURF (Bay, Ess et al. 2008)) in addition to the proposed new contour-based recognition approach. The recognition module could also include an example based learner (e.g. a case-based reasoned) so that the context could be included in the recognition process. For example, common scenes could include "by the beach" where we would expect to have, say, beach balls, or "in the office" scenes, where we would expect to have tables.

The feedback interaction module between the recognition and segmentation module was not implemented, and the type of interaction is yet to be determined. This is should be investigated in the future.

References

Adamek, T. and N. O'Connor (2003). Efficient Contour-based Shape Representation and Matching. MIR'03, Berkeley, California.

Alter, T. D. and R. Basri. (1998). "Extracting Salient Curves from Images: An Analysis of the Saliency Network." International Journal in Computer Vision **27**(1): 51-69.

Ando, S. (2000). "Image field categorization and edge/corner detection from gradient covariance." IEEE Transactions of Pattern Analysis and Machine Intelligence **22**(2): 179-190.

Anzai, A., X. Peng, et al. (2007). "Neurons in monkey visual area V2 encode combinations of orientations." Nature Neuroscience **10**: 1313-1321.

Attneave, F. (1954). "Some informational aspects of visual perception." Psychological Review **61**: 183-193.

Bai, X., Q. Li, et al. (2009). Shape band: A deformable object detection approach. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Miami, Florida, IEEE.

Bai, X., X. Yang, et al. (2008). "Detection and recognition of contour parts based on shape similarity." Pattern Recognition **41**: 2189-2199.

Ballard, D. H. (1981). "Generalizing the Hough transform to detect arbitrary shapes." Pattern Recognition **13**(2): 111-122.

Bay, H., A. Ess, et al. (2008). "SURF: Speeded Up Robust Features." Computer Vision and Image Understanding **110**(3): 346-359.

Belkasim, S. O. (1991). "Pattern recognition with moment invariants – A comparative study and new results." Pattern Recognition **24**(12): 1117-1138.

Belongie, S., J. Malik, et al. (2002). "Shape Matching and Object Recognition Using Shape Contexts." IEEE Transactions of Pattern Analysis and Machine Intelligence **24**(24): 509-522.

Besag, J. (1986). "On the Statistical Analysis of Dirty Pictures." Journal of the Royal Statistical Society Series B **48**(3): 259-302.

Biederman, I. (1987). "Recognition-by-components: a theory of human image understanding. ." Psychological Review **94**(2): 115-147.

Blum, C. (2005). "Ant colony optimization: Introductin and recent trends. ." Physics of Life Reviews **2**(4): 353-373.

Blumberg, J. and G. Kreiman (2010). "How cortical neurons help us see: visual recognition in the human brain." The Journal of Clinical Investigation **120**(9): 3054-3063.

Boykov, Y. and G. Funka-Lea (2006). "Graph Cuts and Efficient N-D Image Segmentation." International Journal of Computer Vision **70**(2): 109-131.

Boykov, Y. and M.-P. Jolly (2001). Interactive Graph Cuts for Optimal Boundary & Region Segmentation of Objects in N-D Images. International Conference on Computer Vision, Vancouver, Canada.

Braje, W. L., D. Kersten, et al. (1998). "Illumination effects in face recognition." Psychobiology **26**(4): 371-380.

Bresson, X., P. Vanderghyenst, et al. (2006). "A Variational Model for Object Segmentation Using Boundary Information and Shape Prior Driven by the Mumford-Shah Functional." International Journal in Computer Vision **68**(2): 145-162.

Brincat, S. L. and C. E. Connor (2006). "Dynamic shape synthesis in posterior inferotemporal cortex." Neuron **49**: 17-24.

Brooks, R. (2003). Robot: The future of Flesh and Machines, Penguin.

Brox, T., M. Rousson, et al. (2003). Unsupervised Segmentation Incorporating Colour, Texture, and Motion. Computer Analysis of Images and Patterns. N. Petkov and M. A. Westenberg, Springer-Verlag. **LNCS 2756:** pp. 353-360.

Bruce, V., E. Hanna, et al. (1992). "The importance of 'mass' in line drawing of faces." Applied Cognitive Psychology **6**: 619-628.

C. Rother, V. Kolmogorov, et al. (2004). "GrabCut: Interactive Foreground Extraction using Iterated Graph Cuts. ." ACM Transactions on Graphics (SIGGRAPH'04).

Cadieu, C., M. Kouh, et al. (2004). "Shape Representation in V4: Investigating Position-Specific Tuning for Boundary Conformation with the Standard Model of Object Recognition." MIT CBCL Memo **241**.

Canny, J. (1986). "A computational approach to edge detection." IEEE Transactions of Pattern Analysis and Machine Intelligence **8**(6): 679-698.

Caselles, V., R. Kimmel, et al. (1997). "Geodesic active contours." International Journal in Computer Vision **22**(1): 61-79.

Chaji, N. and H. Ghassemian (2006). "Texture-Gradient-Based Contour Detection." EURASIP Journal on Applied Signal Processing **2006**(21709): 1-8.

Chan, T., B. Sanberg, et al. (2000). "Active Contours without Edges for Vector-Valued Images." Journal of Visual Communication and Image Representation **11**(2): 130-141.

Chan, T. and L. Vese (2001). "Active Contour Model without Edges." IEEE Transactions on Image Processing **10**: 266-277.

Chen, G. and Y. H. H. Yang (1995). "Edge detection by regularized cubc B-spline fitting." IEEE Transactions of Systems, Man and Cybernetics **25**(4): 636-643.

Cheong, M. and K. S. Loke (2008). An Approach to Texture-Based Image Recognition by Deconstructing Multispectral Co-occurrence Matrices using Tchebichef Orthogonal Polynomials. Proc. ICPR.

Cheong, M. and K. S. Loke (2008). Textile Recognition Using Tchebichef Moments of Co-occurrence Matrices. Proc. 4th International Conference on Intelligent Computing, Shanghai, China.

Cohen, L. (1991). "On active contours models and ballons." Computer Vision Graphics and Image Understanding **53**: 211-218.

Cox, I. J., J. M. Rehg, et al. (1993). "A Bayesian Multiple-Hypothesis Approach to Edge Grouping and Contour Segmentation." International Journal in Computer Vision **11**(1): 5-24.

Cremers, D., T. Kohlberger, et al. (2003). "Shape statistics in kernel space for variational image segmentation." Pattern Recognition **36**(9): 1929-1943.

Cremers, D., M. Rousson, et al. (2007). "A Review of Statistical Approaches to Level Set Segmentation: Integrating Color, Texture, Motion and Shape." International Journal in Computer Vision **72**(2): 195-215.

Cremers, D., C. Schnorr, et al. (2001). Diffusion-Snakes: Combining Statistical Shape Knowledge and Image Information in a Variational Framework. IEEE Workshop on Variational and Levelset Methods**:** 137-144.

Cremers, D., C. Schnorr, et al. (2000). "Diffusion-Snakes Using Statistical Shape Knowledge." Algebraic Frames for the Perception-Action Cycle Lecture Notes in Computer Science **1888**: 164-174.

Cremers, D., F. Tischhauser, et al. (2002). "Diffusion Snakes: Introducing statistic shape knowledge into Mumford-Shah functional." International Journal on Computer Vision **50**(3): 295-313.

Dalal, N. and B. Triggs (2005). Histograms of Oriented Gradients for Human Detection. IEEE Conference on Computer Vision and Pattern Recognition, San Diego, California.

Datta, R., D. Joshi, et al. (2008). "Image Retrieval: Ideas, Influences, and Trends of the New Age." ACM Computing Surveys **40**(2): 1-60.

Davies, G., H. D. Ellis, et al. (1978). "Face recognition accuracy as function of mode of representation." Journal of Applied Psychology **63**: 180-187.

Diamond, P. E. and S. Carey (1986). "Why faces are and are not special: An effect of expertise." Journal of Experimental Psychology: General **115**(2): 107-117.

DiCarlo, J. J. and D. D. Cox (2007). "Untangling invariant object recognition." Trends in Cognitive Science **11**(8): 333-341.

Dimov, D. (2003). Fast, Shape Based Image Retrieval. Proceedings of the 4th International Conference on Computer Systems and Technologies. Rousse, Bulgaria**:** 296-302.

Donoho, D. L. and X. Huo (2000). Beamlet pyramids: a new form of multiresolution analysis, suited for extracting lines, curves and objects from very noisy image data. Wavelet applications in signal and image processing VIII SPIE, , San Diego, CA.

Donoho, D. L. and X. Huo (2001) "Applications of beamlets to detection and extraction of lines, curves and objects in very noisy image Nonlinear Signal and Image Processing (NSIP)." Nonlinear Signal and Image Processing (NSIP), 4119-4149.

Donoho, D. L. and X. Huo (2002). Beamlets and Multiscale Image Analysis. Proc. Multiscale and Multiresolution Methods LNCSE. T. J. Barth, T. Chan and R. Haimes. **20:** 149–196.

Dorigo, M. and T. Stützle (2004). Ant Colony Optimization, MIT Press.

Estrada, F. J. (2005). Advances in Computational Image Segmentation and Perceptual Grouping. Phd, University of Toronto.

Estrada, F. J. and A. D. Jepson (2006). Robust Boundary DetectionWith Adaptive Grouping. Computer Vision and Pattern Recognition Workshop**:** 184 - 184.

Feldman, J. and M. Singh (2005). "Information Along Contours and Object Boundaries." psychological Review **112**(1): 263-252.

Felzenszwalb, P. and D. McAllester (2006). A Min-Cover Approach for Finding Salient Curves. 5th IEEE Computer Society Workshop on Perceptual Organization in Computer Vision (CVPR 2006), New York.

Felzenszwalb, P. and J. Schwart (2007). Hierarchical Matching of Deformable Shapes. IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Fenske, M. J., E. Aminoff, et al. (2006). "Top-down facilitation of visual object recognition: object-based and context-based contributions." Progress in Brain Research **155**(3-21).

Ferrari, V., F. Jurie, et al. (2007). Accurate object detection with deformable shape models learnt from images. IEEE International Conference on Computer Vision and Pattern Recognition, Minneapolis, US, IEEE.

Ferrari, V., F. Jurie, et al. (2010). "From Images to Shape Models for Object Detection." International Journal in Computer Vision **87**(3).

Ferrari, V., T. Tuytelaars, et al. (2006). Object Detection by Contour Segments Networks. European Conference on Computer Vision, Graz, Austria.

Fischler, M. A. and R. C. Bolles (1981). "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography." Commun. ACM **24**(6): 381-395.

Folkers, A. and H. Samet (2002). Content-based Image Retrieval Using Fourier Descriptors on a Logo Database. Proceedings of the 16th Conference on Pattern Recognition. Quebec City, Canada. **III:** 521-524.

Folsom, T. C. and R. B. Pinter (1998). "Primitive features by steering, quadrature, and scale." IEEE Transactions of Pattern Analysis and Machine Intelligence **20**(11): 1161-1173.

Frei, W. and C.-C. Chen (1977). "Fast boundary detection: a generalization and a new algorithm." IEEE Transactions on Computers **26**(10): 988-998.

Freund, Y. and L. Mason (1999). The Alternating Decision Tree Algorithm. The 16th International Conference on Machine Learning.

Friedman, J., T. Hastie, et al. (1998). "Additive Logistic Regression: a Statistical View of Boosting." Annals of Statistics **28**(2000): 337-407.

Friedrich, F. (2005). Complexity Penalized Segmentation in 2D: Efficient Algorithms and Approximation Properties. PhD, Technical University of Munich.

Friedrich, F., L. Demaret, et al. (2007). "Efficient moment computation over polygonal domains with an application to rapid wedgelet approximation." SIAM Journal of Scientific Computing **29** (2): 842 - 863.

Führ, H., L. Demaret, et al. (2006). Beyond wavelets: New image representation paradigms. Document and Image Compression. M. Barni, CRC Press.

Geman, S. and C. Graffigne (1986). Markov Random Field Image Models and Their Applications to Computer Vision. Proceedings of the International Congress of Mathematicians. Berkeley, California, USA.**:** 1496-1517.

Gibson, J. J. (1950). The Perception of the Visual World. Boston, Houghton Mifflin.

Gonzales, R. C. and R. E. Woods (2002). Digital Image Processing, Prentice-Hall.

Graboi, D. and J. Lisman (2003). "Recognition by Top-Down and Bottom-Up Processing in Cortex: The Control of Selective Attention." Journal of Neurophysiology **90**: 798-810.

Grill-Spector, K., T. Kushnir, et al. (1998). "A Sequence of Object-Processing Stages Revealed by fMRI in the Human Occipital Lobe." Human Brain Mapping **6**: 316-328.

Grill-Spector, K. and R. Sayres (2008). "Object Recognition - Insights From Advances in fMRI Methods." Current Directions in Psychological Science **17**(2).

Harris, C. and M. Stephens (1988). A combined corner and edge detector. Alvey Vision Conference**:** 453-507.

Hartmut Fuhr, L. D., Felix Friedrich (2006). Beyond wavelets: New image representation paradigms. Document and Image Compression. M. Barni, CRC Press.

Heath, M., S. Sarkar, et al. (1996). Comparison of Edge Detectors: A Methodology and Initial Study. CVPR'96.

Hegde, J. and D. C. Van Essen (2000). "Selectivity for complex shapes in primate visual area V2." The Journal of Neurophysiology **20**(RC61): 1-6.

Hegde, J. and D. C. Van Essen (2003). "Strategies of shape representation in macaque visual area V2." Visual Neuroscience **20**(3): 313-328.

Hidayat, R. and R. Green (2009). Real-time texture boundary detection from ridges in the standard deviation space. British Machine Vision Conference, London.

Hildreth, E. C. (1983). "The detection of intensity changes by computer and biological vision systems." Computer Vision, Graphics and Image Processing **22**(1): 1-27.

Hoffman, D. D. and M. Singh (1997). "Salience of visual parts." Cognition **63**: 29-78.

Hsu, H. S. (1993). "Moment preserving edge detection and its application to image data compression." Opt. Eng **32**(7): 1596-1608.

Hu, M. (1962). "Visual Pattern Rcognition by Moment Invariants." IRE Trans. Information Theory **IT-8**: 179-187.

Hubel, D. and T. Wiesel (1962). "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex." Journal of Physiology (London) **160**: 106-154.

Hung, C. P., G. Kreiman, et al. (2005). "Fast Readout of Object Identity from Macaque Inferior Temporal Cortex." Science **310**: 863-865.

Itti, L. and C. Koch (1999). Learning to Detect Salient Objects in Natural Scenes Using Visual Attention. Image Understanding Workshop.

Jiang, X. (2007). "Extracting image orientation feature by using integration operator." Pattern Recognition **40**: 705-717.

Joshi, G. D. and J. Sivaswamy (2005). A Multiscale Approach to Salient Contour Extraction. Proceedings of International Conference on Cognition and Recognition. Mysore, India.

Joshi, G. D. and J. Sivaswamy (2006). A Simple Scheme for Contour Detection. Proceedings of International Conference on Computer Vision and Applications (VISAP 2006). Setubal.

Juan, O. and Y. Boykov (2006). Active Graph Cuts. IEEE International Conference on Computer Vision and Pattern Recognition. New York, USA. **I:** 1023-1029.

Julesz, B. (1971). Foundations of Cyclopean Perception. Chicago, The University of Chicago Press.

Kadir, T. and M. Brady (2001). "Scale, saliency and image description." International Journal in Computer Vision **45**(2): 83-105.

Kanizsa, G. (1979). Organization in Vision: Essays on Gestalt Perception. New York, Praeger.

Kass, M., A. Witkin, et al. (1988). "Snakes: active contour models." International Journal of Computer Vision: 321-331.

Kichenassamy, S., A. Kumar, et al. (1995). "Gradient flows and geometric active contour models." IEEE International Conference in Computer Vision: pp. 810-815.

Kim, J. G., I. Biederman, et al. (2009). "Adaptation to objects inthe lateral occipital complex (LOC): Shape or semantics." Vision Reseach **49**: 2297-2305.

Kohli, P. and P. H. S. Torr (2007). "Dynamic Graph Cuts for Efficient Inference in Markov Random Fields." IEEE Transactions of Pattern Analysis and Machine Intelligence **29**(12): 2079-2088.

Kourtzi, Z. and N. Kanwisher (2001). "Processing of perceived shape vs contours in the human lateral occipital complex." Journal of Vision **1**(3).

Kpalma, K., M. Yang, et al. (2008). Planar Shapes Descriptors Based on the Turning Angle Scalogram. ICIAR '08 Proceedings of the 5th international conference on Image Analysis and Recognition, Springer-Verlag.

Kristjansson, A. and P. U. Tse (2001). "Curvature discontinuities are cues for rapid shape analysis." Perception & Psychophysics **3**(63): 390-403.

L. Kotoulas, I. A. (2005). Image Analysis Using Moments. Proc. of ICTA'05, Greece.

L. M. Luo, X. H. X., X. D. Bao (1994). "A modified moment-based edge operator for rectangular pixel image", IEEE Trans. Circuits Systems Video Technology." **4**: 552-554.

L.R. Ford, S. and E. Fulkerson (1962). Flows in Networks. Princeton, New Jersey, Princeton Univ Press.

Latecki, L. J., R. Lakaemper, et al. (2000). Shape descriptors for non-rigid shapes with a single closed contour. IEEE Conference on Computer Vision and Pattern Recognition**:** 424-429.

Leibe, B., A. Leonardis, et al. (2004). Combined Object Categorization and Segmentation with Implicit Shape Model. ECCV '04 Workshop on Statistical Learning in Computer Vision. Praque.

Leventon, M. E., W. E. Grimson, et al. (2000). Statistical shape influence in geodesic active contours. IEEE Conference on Computer Vision and Pattern Recognition.

Lew, M. S., N. Sebe, et al. (2006). "Content-based Multimedia Information Retrieval: State of the Art and Challenges." ACM Transactions on Multimedia Computing, Communications and Applications **2**(1): 1-19.

Li, C., C. Xu, et al. (2005). Level Set Evolution Without Re-initialization: A New Variational Formulation. IEEE Conference on Computer Vision and Pattern Recognition. **1:** pp. 430-436.

Li, S. Z. (1994). Markov Random Field Models in Computer Vision. Proceedings of the Third European Conference on Computer Vision. **LNCS 801:** 361-370

Li, Y., J. Sun, et al. (2004). Lazy Snapping. SIGGRAPH

Lin, H.-J., Y.-T. Kao, et al. (2004). A Study of shape-Based Image Retrieval. Proceedings of the 24th International Conference on Distributed Computing Systems Workshops(ICDCSW'04). **7:** 118-123.

Liu, T., J. Sun, et al. (2007). Finding important areas in images using conditional random field. IEEE Computer Vision and Pattern Recognition Conference. Minneapolis, MN.

Lo, C. H. and H. S. Don (1989). "3D moment forms: Their construction and application to object identification and positioning." IEEE Trans. Pattern Anal. Mach. Intell. **11**: 1053-1064.

Loke, K. S. (2009). An Approach to Textile Recognition. Pattern Recognition. P.-Y. Yin, In-TECH

Loke, K. S. and M. Cheong (2009). Efficient Textile Recognition via Decomposition of Co-occurrence Matrices. IEEE International Conference on Signal & Image Processing ICSIPA '09. Kuala Lumpur.

Loncaric, S. (1998). "A survey of shape analysis techniques." Pattern Recognition **31**(8): 983-1001.

Lowe, D. G. (2004). "Distinctive Image Features from Scale-Invariant Keypoints." International Journal of Computer Vision **60**(2): 91-110.

Lu, C., L. J. Latecki, et al. (2009). Shape guided contour grouping with particle filters. IEEE International Conference on Computer Vision, Kyoto, Japan, IEEE.

Lu, D. S. and C. C. Chen (2008). "Edge detection improvement by ant colony optimization." Pattern Recog. Letters **29**: 416-425.

M.Bar, K. S. Kassma, et al. (2006). "Top-down facilitation of visual recognition." Proc. Natl. Acad. Sci USA **103**(2): 449-454.

Macanu, I. (2007). Image Retrieval by Shape Based on Contour Technique A Comparative Study. 4th International Symposium on Applied Computational Intelligence and Informatics. Timisoara, Romania**:** 219-223.

Malik, J., S. Belongie, et al. (2001). "Contour and texture analysis for image segmentations." International Journal of Computer Vision **43**(1): 7-27.

Malik, J., S. Belongie, et al. (1999). Textons, Contours and Regions: Cue Integration in Image Segmentation. IEEE International Conference on Computer Vision. Corfu, Greece**:** 918 - 925.

Malladi, R., J. A. Sethian, et al. (1995). "Shape modeling with front propagation: a level set approach." IEEE Transactions Pattern Analysis Machine Intelligence **17**(2): 158-175.

Manjunath, B. S. and W.-Y. Ma (1996). "Texture features for browsing and retrieval of image data." IEEE Transactions of Pattern Analysis and Machine Intelligence **18**(8): 837-842.

Marr, D. (1982). Vision: A computation investigation into human representation and processing of visual information, W.H. Freeman.

Marr, D. and E. C. Hildreth (1980). "Theory of edge detection." Proceedings of the Royal Society of London. Series B. Biological sciences **207**(1167): 187-217.

Marr, D. and H. K. Nishihara (1978). "Representation and Recognition of the Spatial Organization of Three-Dimensional Shapes." Proceedings of the Royal Society of London. Series B, Biological Sciences **200**(1140): 269-294.

Martens, J.-B. (1997). "Local orientation analysis in images by means of the Hermite transform." IEEE Transactions on Image Processing **6**(8): 1103-1116.

Mata, J., O. Chum, et al. (2002). Robust wide-baseline stereo from maximally stable extremal regions. Proceedings of the British Machine Vision Conference, Cardiff, UK.

Meer, P. and B. Georgescu (2001). "Edge detection with embedded confidence." IEEE Transactions of Pattern Analysis and Machine Intelligence **23**(12): 1351-1365.

Mikolajczyk, K. and C. Schmid (2001). Indexing based on scale invariant interest points. Proceedings of the 8th International Conference on Computer Vision, Vancouver, Canada, IEEE Computer Society.

Mikolajczyk, K. and C. Schmid (2005). "A performance evaluation of local descriptors." IEEE Transactions on Pattern Analysis and Machine Intelligence **27**(10): 1615-1630.

Mokhtarian, F. and S. Abbasi (1999). Curvature scale space for shape similarity retrieval under affine transforms. International Conference on Computer Analysis of Images and Patterns**:** 65-72.

Mokhtarian, F. and S. Abbasi (1999). Shape-based indexing using curvature scale space with affine curvature. European workshop on content-based multi-media indexing**:** 255-262.

Moosmann, F., D. Larlus, et al. (2006). Learning Saliency Maps for Object Categorization. International Workshop on The Representation and Use of Prior Knowledge in Vision. Gratz.

Mori, G., S. Belongie, et al. (2001). Shape contexts enable efficient retrieval of similar shapes. International Conference on Computer Vision and Pattern Recognition.

Mori, G., S. Belongie, et al. (2005). "Efficient Shape Matching Using Shape Contexts." IEEE Transactions of Pattern Analysis and Machine Intelligence **27**(11): 1832-1837.

Morrone, M. C. and D. C. Burr (1988). "Feature detection in human vision: a phase-dependent energy model." Proceedings of the Royal Society of London. Series B. Biological sciences **235**(1280): 221-245.

Mukundan, R. (2006). Transform Coding Using Discrete Tchebichef Polynomials. Proc. of IASTED Intnl. Conf. on VIIP, Palma Mallorca, Spain.

Mukundan, R., S. H. Ong, et al. (2001). "Image Analysis by Tchebichef Moments." IEEE Transactions of Image Processing **10**(9): 1357 - 1364.

Mukundan, R., S. H. Ong, et al. (2001). "“Image Analysis by Tchebichef Moments”." IEEE Trans. Image Proc **10**(9).

Mukundan, R. and K. R. Ramakrishnan (1998). Moment Functions in Image Analysis Theory and Applications. Singapore, World Scientific.

Mumford, D. and J. Shah (1985). Boundary detection by minizing functionals I. IEEE Conference on Computer Vision and Pattern Recognition, San Fransisco.

Murphy, T. M. and L. H. Finkel (2007). "Shape representation by a network of V4-like cells." Neural Networks **20**: 851-867.

Opelt, A., A. Pinz, et al. (2006). A Boundary-Fragment Model for Object Detection. European Conference on Computer Vision.

Osher, S. and N. Paragios, Eds. (2003). Geometric level set methods in imaging, vision, and graphics, Springer-Verlag.

Osher, S. and J. A. Sethian (1988). "Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations." Journal of Commputational Physics **79**(1): 12-49.

Palmer, S. E. (1975). "The effects of contextual scenes on the identification of objects." Memory and Cognition **3**: 519-526.

Papademetriou, R. C. (1992). Reconstruction with moments. Proceedings of 11th International Conference of Pattern Recognition.

Papari, G., P. Campisi, et al. (2007). Multilevel Surround Inhibition: A Biologically motivated Contour Detector. Proceedings SPIE 2007 Image processing: Algorithm and Systems San Jose, CA. **649702:** 1 -11.

Papari, G., P. Campisi, et al. (2006). A Multiscale Approach to Contour Detection by Texture Suppression. Proceeding SPIE 2006 Image processing: Algorithm and Systems San Josè, CA. **6064A:** 1 - 12.

Papari, G., P. Campisi, et al. (2007). "A Biologically Motivated Multiresolution Approach to Contour Detection." EURASIP Journal on Advances in Signal Processing **2007**(71828): 119-119.

Paragios, N. (2005). Curve Propagation, Level Set Methods and Grouping. Mathematical Models in Computer Vision: The Handbook. Y. C. Nikos Paragios, Olivier Faugeras, Springer.

Paragios, N. and R. Deriche (1999). Geodesic Active Contours for Supervised Texture Segmentation. IEEE International Conference on Computer Vision. **2** 926-932.

Pasupathy, A. and C. E. Connor (1999). "Responses to contour features in macaque area V4." The Journal of Neurophysiology **82**(5): 2490-2502.

Pasupathy, A. and C. E. Connor (2001). "Shape representation in area V4: Position-specific tuning for boundary conformation." The Journal of Neurophysiology **86**(5): 2505-2519.

Perez, P., A. Blake, et al. (2001). JetStream: Probabilistic Contour Extraction with Particles. Proceedings of the Eighth IEEE International Conference on Computer Vision, Vancouver, Canada.

Perona, P. and J. Malik (1990). "Scale-space and edge detection using anisotropic diffusion." IEEE Transactions of Pattern Analysis and Machine Intelligence **12**(7): 629-639.

Perrett, P. and M. Oram (1993). "Neurophysiology of shape processing." Image and Vision Computing **11**: 317-333.

Peterson, M. A. (1994). "Object Recognition Processes Can and Do Operate Before Figure-Ground Organization." Current Directions in Psychological Science **3**(105-111).

Peterson, M. A. and B. S. Gibson (1991). "The initial identificaiton of figure-ground relationship." Bulletin of the Psychonomic Society **29**: 199-202.

Peterson, M. A. and B. S. Gibson (1994). "Must Figure-Ground Organization Precede Object Recognition? An Assumption in Peril." Psychological Science **5**(5): 253-259.

Pfahringer, B., G. Holmes, et al. (2001). Optimizing the induction of Alternating Decision Trees. Fifth Pacific Asia Conference on Advances in Knowledge Discovery and Data Mining.

Pinto, N., D. D. Cox, et al. (2008). "Why is Real-World Visual Object Recognition Hard?" PLoS Computational Biology **4**(1): pp. 151-156.

Pitas, I. (1988). "Markovian Image Models for Image Labelling and Edge Detection." Signal Processing **15**: 365-374.

Pongpiyapaiboon, P. (2005). Development of Efficient Algorithm for Geometric Representation based on Arclet Decomposition Master, Technische Universität München.

Ramachandran, V. S. and D. Rogers-Ramachandran (2008). Seeing Is Believing. Scientific American Special. **18:** 26-28.

Ravishankar, S., A. Jain, et al. (2008). Multi-stage Contour Based Detection of Deformable Objects. Proceedings of the 10th European Conference on Computer Vision, Marseille, France.

Reicher, G. M. (1969). "Perceptual recognition as a function of meaninfulness of stimulus material." Journal of Experimental Psychology **81**: 274-280.

Ren, X., C. Fowlkes, et al. (2005). Mid-level Cues Improve Boundary Detection. Report No:UCB/CSD-5-1382. Berkeley, University of California.

Ren, X., C. Fowlkes, et al. (2005). Scale-Invariant Contour Completion using Conditional Random Fields. ICCV '05. Beijing. **2:** 1214-1221.

Ren, X., C. Fowlkes, et al. (2006). Figure/Ground Assignment in Natural Images. ECCV '06. Graz. **2:** 614-627.

Ren, X., C. C. Fowlkes, et al. (2005). Scale-Invariant Contour Completion Using Conditional Random Fields. Tenth IEEE International Conference on Computer Vision (ICCV '05).

Ren, X. and J. Malik (2002). "The Ecological Statistics of Good Continuation: Multi-scale Markov Models for Contours." Journal of Vision **2**(7): 708a.

Ren, X. and J. Malik (2002). A Probabilistic Multi-scale Model for Contour Completion Based on Image Statistics. ECCV'02. Copenhagen. **1:** 312-327.

Resnikoff, H. L. (1985). The illusion of reality: Topics innformation science. New York, Springer-Verlag.

Riesenhuber, M. and T. Poggio (1999). "Hierarchical models of object recognition in cortex." Nature Neuroscience **2**(11): 1019-1025.

Rousson, M., T. Brox, et al. (2003). Active Unsupervised Texture Segmentation on a Diffusion Based Feature Space. IEEE Computer Society Conference on Computer Vision and Pattern Recognition **2:** 699-704.

Rumelhart, D. E. and J. L. McClelland (1982). "An interactive activation model of context effects in letter perception." Psychological Review **89**: 60-94.

Rumelhart, D. E. and J. L. McClelland (1986). Parallel distributed processing: Explorations in the microstructure of cognition. Vol. 1: Foundations. Cambridge, MIT Press.

Rusinol, M., P. dosch, et al. (2007). "Boundary Shape Recognition Using Accumulated Length and Angle Information." Lecture Notes in Computer Science **4478**: 210-217.

Ruzon, M. A. and C. Tomasi (2001). "Edge,Junction, and Corner Detection Using Color Distributions." IEEE Transactions of Pattern Analysis and Machine Intelligence **23**(11): 1-16.

Sanocki, T., K. W. Bowyer, et al. (1998). "Are Edges Sufficient for Object Recognition." Journal of Experimental Psychology: Human Perception and Performace **24**(1): 340-349.

Sarfraz, M. and A. Ridha (2007). Content-based Image Retrieval using Multiple Shape Descriptors. IEEE International Conference on Computer Systems and Applications. Amman**:** 730-737.

Scassellati, B., S. Alexopoulos, et al. (1994). Retrieving images by 2D shape: a comparison of computation methods with perceptual judgements. Storage and Retrieval for Image and Video Databases II SPIE Conference Proceedings, San Jose, CA.

Schwarts, E. L., R. Desimone, et al. (1983). "Shape recogntion and inferior temporal neurons." Proc. Natl. Acad. Sci USA **80**: 5776-5778.

Schwartz, E. L., R. Desimone, et al. (1983). "Shape recognition and inferior temporal neurons." Proc. Natl. Acad. Sci. USA **80**: 5776-5778.

See, K. W., K. S. Loke, et al. (2007). "Image Reconstruction using Various Discrete Orthogonal Polynomials in Comparison with DCT." Journal of Applied Mathematics and Computation **193**(2): 346-359.

Serre, T., L. Wolf, et al. (2007). "Robust Object Recognition with Cortext-Like Mechanisms." IEEE Transactions on Pattern Analysis and Machine Intelligence **29**(3): 411-426.

Sethian, J. A. (1996). Level set methods: evolving interfaces in geometry, fluid mechanics, computer vision and materials sciences, Cambridge.

Sethian, J. A. (1999). Level Set Methods and Fast Marching Methods: Evolging Interfaces in Computational Geometry, Fluid Mechanics, Computer Vsion and Material Sciences. Cambridge, U.K., Cambridge Univ. Press.

Shah, J. and D. Mumford (1985). Boundary Detection by Minimizing Functionals, I. Proc. IEEE Conf. on Computer Vision and Pattern Recognition, San Francisco.

Shannon, C. E. (1948). "A Mathematical Theory of Communication." Bell System Technical Journal **27**: 379-423 & 623-656.

Shashua, A. and S. Ullman (1988). Structural Saliency: the Detection of Globally Salient Structures Using a Locally Connected Network. Proc. of the International Conference on Computer Vision Tarpon Springs, Florida.**:** 321-327.

Sheng, C. and Y. Xin (2005). Shape Image Retrieval Using Elastic Matching Combined with Snake Model. International Conference on Image Analysis and Processing. Cagliari, Italy. **LNCS 3617:** 511-518.

Shi, J. and J. Malik (2000). "Normalized Cuts and Image Segmentation." IEEE Transactions of Pattern Analysis and Machine Intelligence **22**(8): 888-905.

Shotton, J., A. Blake, et al. (2008). "Multi-Scale Categorical Object Recognition Using Contour Fragments." IEEE Transactions of Pattern Analysis and Machine Intelligence.

Suri, J. S., K. Liu, et al. (2002). "Shape Recovery Algorithms Using Level Sets in 2-D/3-D Medical Imagery: A State-of-the-Art Review." IEEE Transactions on Information Technology in Biomedicine **6**(1): pp. 8-28.

Tanaka, J. W. and M. J. Farah (1993). "Parts and wholes in face recognition." Quarterly Journal of Experimental Psychology **46**(2): 225-245.

Teague, M. R. (1980). "Image analysis via the general theory of moments." J. Opt. Soc. Am(70): 920-930.

Thayananthan, A., B. Stenger, et al. (2003). Shape context and chamfer matching in cluttered scenes. Proceedings of the International Conference of Computer Vision and Pattern Recognition. Madison, USA**:** 127-133.

Thorpe, S., D. Fize, et al. (1996). "Speed of processing in human visual system." Nature **381**(520-522).

Tsao, D. Y., W. A. Freiwald, et al. (2006). "A cortical region consisting entirely of face-selective cells." Science **311**(5761): 670-674.

Tuceryan, M. (1994). "Moment-based texture segmentation." Pattern recognition Letter **15**: 115-123.

Tuytelaars, T. and K. Mikolajczyk (2007). "Local Invariant Feature Detectors: A Survey." Foundations and Trends in Computer Graphics and Vision **3**(3): 177-280.

Ullman, S. (1996). High Level Vision, M.I.T. Press.

Ungerleider, L. G. and M. Mishkin, Eds. (1982). Analysis of Visual Behavior, MIT Press.

Vailaya, A. (1996). Shape-based image retrieval. Masters, Michigan State University.

Vecera, S. P. and M. J. Farah (1997). "Is visual Image segmentation a bottom-up or an interactive process." Perception & Psychophysics **59**(8): 1280-1296.

Vecera, S. P. and R. C. O'Reilly (2000). "Graded Effects in Hierarchical Figure-Ground Organization: Reply to Peterson(1999)." Journal of Experimental Psychology: Human Perception and Performance **26**(3): 1221-1231.

Viola, P. and M. J. Jones (2004). "Robust Real-Time Face Detection." International Journal of Computer Vision **57**(2): 137-154.

Walther, D. and C. Koch (2006). "Modeling attention to salient proto-objects." Neural Networks **19**: 1395-1407.

Walther, D., U. Rutishauser, et al. (2005). "Selective visual attention enables learning and recognition of multiple objects in cluttered scenes." Computer Vision and Image Understanding **100**: 41-63.

Wang, J., K. W. Bowyer, et al. (1998). The effect of edge strength on object recognition from edge images. International Conference on Image Processing.

Wang, L., J. Shi, et al. (2005). Object Detection Combining Recognition and Segmentation. Eighth Asian Conference on Computer Vision (ACCV), 2007 Tokyo, Japan.

Weijer, J. v. d., T. Vevers, et al. (2005). "Edge and Corner Detection by Photometric Quasi-Invariants." IEEE Transactions of Pattern Analysis and Machine Intelligence **27**(4).

Weiss, Y. (2000). "Correctness of local probability propagation in graphical models with loops." Neural Computation **12**(1): 1-41.

Wikipedia. (2011). "Horse markings." Retrieved April 11, 2011, from http://en.wikipedia.org/w/index.php?title=Horse_markings&oldid=421176368.

Williams, D. J. and M. Shah (1990). "Edge Contours Using Multiple Scales." Computer Vision, Graphics and Image Processing **51**: 256-274.

Winter, J. D. and J. Wagemans (2008). "Perceptual saliency of points along the contour of everyday objects: A large-scale study." Perception & Psychophysics **1**(70): 50-64.

Wu, Z. and R. Leahy (1993). "An Optimal Graph Theoretic Approach to Data Clustering: Theory and Its Application to Image Segmentation." IEEE Transactions on Pattern Analysis and Machine Intelligence **15**(11): 1101-1113.

Xu, C. and J. Prince (1998). "Snakes, shapes, and gradient vector flow." IEEE Transactions Image Processing **7**(3): 339-348.

Y.LeCun, L. Bottou, et al. (1998). "Gradient-Based Learning Applied to Document Recognition." Proceedings of the IEEE **86**(11): 2278-2324.

Yahiaoui, I., N. Herve, et al. (2006). Shape-based image retrieval in botanical collections. 7th Pacific Rim Conference on Multimedia. Hangzhou, China, Springer **4261:** 357-364.

Yap, P. T., R. Paramesran, et al. (2003). "Image Analysis by Krawtchouk Moments." IEEE Trans. Image Proc. **12**(11): 1367-1377.

Yezzi, A., Kichenassamy, S , Kumar, A , Olver, P , and Tannenbaum, A. (1997). "A geometric snake model for segmentation of medical imagery " IEEE Transactions on Medical Imaging **16**: 199-209.

Yin, R. K. (1969). "Looking at upside-down faces." Journal of Experimental Psychology **81**(1): 141-145.

Young, A. W., D. Hellawell, et al. (1987). "Configurational information in face perception." Perception **16**(6): 747-759.

Yu, S. X. and J. Shi (2001). Understanding Popout through Repulsion. International IEEE Conference on Computer Vision and Pattern Recognition. **2:** 752-757.

Zhang, D. S. and G. Lu (2002). Generic Fourier Decriptors for Shape-based Image Retrieval. IEEE International Conference on Multimedia and Expo. Lausanne, Switzerland. **1:** 425-428.

Zhou, C. and B. W. Mel (2008). "Cue combination and color edge detection in natural scenes." Journal of Vision **8**(4): 1-25.

Zhou, J., H. Shu, et al. (2005). Image Analysis by Discrete Orthogonal Hahn Moments. ICIAR 2005.

Zhu, Q., L. Wang, et al. (2008). Contour context selection for object detection: A set-to-set contour matching approach. European Conference on Computer Vision.

Ziou, D. and S. Tabbone (1993). "A Multi-Scale Edge Detector." Pattern Recognition **26**(9): 1305-1314.

Ziou, D. and S. Tabbone (1998). "Edge Detection Techniques An Overview." International Journal of Pattern Recognition and Image Analysis **8**: 537-559.

Zusne, L. (1970). Visual perception of form. New York, Academic Press.