

Image Segment Processing for Analysis and Visualization

by

Darren T. MacDonald

Thesis submitted to the
Faculty of Graduate and Postdoctoral Studies
In partial fulfillment of the requirements
For the MCS degree in
Computer Science

School of Information Technology and Engineering
Faculty of Engineering
University of Ottawa

© Darren T. MacDonald, Ottawa, Canada, 2008

Abstract

This thesis is a study of the probabilistic relationship between objects in an image and image appearance. We give a hierarchical, probabilistic criterion for the Bayesian segmentation of photographic images. We validate the segmentation against the Berkeley Segmentation Data Set, where human subjects were asked to partition digital images into segments each representing a ‘distinguished thing’. We show that there exists a strong dependency between the hierarchical segmentation criterion, based on our assumptions about the visual appearance of objects, and the distribution of ground truth data. That is, if two pixels have similar visual properties then they will often have the same ground truth state. Segmentation accuracy is quantified by measuring the information cross-entropy between the ground truth probability distribution and an estimate obtained from the segmentation. We consider the proposed method for estimating joint ground truth probability to be an important tool for future image analysis and visualization work.

Extended Abstract

Digital cameras are one of the most inexpensive and readily available automatic sensors and the quantity of information they provide is enormous. This thesis is a study of the probabilistic relationship between objects in an image and image appearance. We give a hierarchical, probabilistic criterion for the Bayesian segmentation of photographic images combining the visual cues of size, colour, luminance, and shape. In order to facilitate object-based operations on images the segmentation should be as faithful as possible to the organization of physical objects in the image. Therefore we validate the segmentation against the Berkeley Segmentation Data Set compiled by Martin, Tal, Fowlkes and Mallows. Human subjects were asked to partition digital images into segments, each representing a ‘distinguished thing’. We show that there exists a strong dependency between the hierarchical segmentation criterion, based on our assumptions about the visual appearance of objects, and the distribution of ground truth data.

This is significant because it means an image segmentation can be used to predict the distribution of ‘distinguished things’ in an image. Segmentation accuracy is quantified by measuring the information cross-entropy between the ground truth probability distribution and an estimate obtained from the segmentation. Other ground truth based evaluation methods exist, however the benefit of our approach is that the cross-entropy unit of measure relates directly to the conceptual computer vision application of ‘describing’ the spatial extent of a ground truth segment. If the segmentation matches closely to the ‘distinguished things’ in the image then it will be particularly efficient for this task. After training the system we obtain a compression rate of 2 per cent over a method that disregards pixel correlations.

A concise and simple description of objects is important for the efficiency and robustness of computer vision applications. Moreover from an information theoretical perspective a concise description demonstrates an accurate understanding of the underlying distribution, in this case, of objects in the scene. We consider the proposed method for estimating joint ground truth probability to be an important tool for future image analysis and visualization work.

Acknowledgements

Thanks to my thesis supervisor Dr. Jochen Lang, whose expertise and resources made this thesis possible.

To my loving family, I owe you everything. Many thanks to Lindsay and Riley, who shared in my suffering - I share this accomplishment with you. Thanks to my mother for teaching me to use my imagination and to my father and brother for teaching me how to build things. To Alexander, our newest member: a person can build their vision.

Gup, thank you for your unqualified endorsement.

This research was funded in part by the Natural Sciences and Engineering Research Council of Canada (NSERC) and by the Government of Canada.

Contents

1	Introduction	1
1.1	Contributions	5
1.2	Summary	6
2	Probabilistic Graphical Models in Image Segmentation	8
2.1	Variational and Probabilistic Segmentation Criteria	8
2.2	Models of Image Formation	10
3	A Hierarchical Image Segmentation Algorithm	15
3.1	A Hierarchical Segmentation Criterion	15
3.1.1	A Hierarchical Model for Segment Size	16
3.1.2	A Hierarchical Model for Segment Mean Luminance	17
3.1.3	A Hierarchical Model for Segment Mean Colour	19
3.1.4	A Descriptive Language for Segment Perimeter	19
3.2	Computing the Segmentation	20
4	Related Work on Methods for Segmentation Evaluation	23
4.1	Approaches to Segmentation Evaluation	24
4.2	Measures of Discrepancy for Empirical Evaluation	26
4.2.1	Partition Consensus Methods	26
4.2.2	The Consensus of Hierarchies	28
5	Segment Description: An Efficient Task Based Segmentation Evaluation	31
5.1	An Information Theoretic Measure for Discrepancy	31
5.2	A Hierarchical Model for Ground Truth Segments	35
5.3	Computing the Segment Description Length	38

6	Experimentation	42
6.1	Evolving the Descriptive Language for Reduced Description Length . . .	42
6.2	Results	45
7	Applications	53
7.1	HSVGen: A Hierarchical Bitmap-to-SVG Image Converter	53
7.2	SAFE: A Semi-Automatic Foreground Extractor	56
8	Conclusions and Future Work	60

List of Figures

1.1	Image Segmentation	2
2.1	Probabilistic Graphical Models of Image Formation	12
3.1	The Beta Distribution	18
3.2	The Gaussian Distribution	18
3.3	The Region Merging Algorithm	22
5.1	The intersection of subsegments and ground truth: Size invariance	36
5.2	The intersection of subsegments and ground truth: Detail	37
5.3	The Segment Description Algorithm	41
6.1	Hierarchical Segmentation Results	47
6.2	Compression of Ground Truth	48
6.3	Progress of Solutions	48
6.4	Hierarchical segmentation versus ground truth segmentation: Pyramid .	49
6.5	Hierarchical segmentation versus ground truth segmentation: Lion	50
6.6	Hierarchical segmentation versus ground truth segmentation: Monk . . .	51
6.7	Hierarchical segmentation versus ground truth segmentation: Iceberg . .	52
7.1	HSVGEN: a raster-to-vector conversion program.	54
7.2	SVG4Blind: a tactile image explorer.	54
7.3	A scalable vector graphic (SVG) with interactive level-of-detail.	55
7.4	A scalable vector graphic (SVG): Detail	55
7.5	Background Subtraction with SAFE	58
7.6	Foreground image composite	59

Chapter 1

Introduction

Analogously to the way sight allows humans to perceive their environment, computer vision is the automatic processing of digital imagery in order to acquire some information about the physical world. Digital cameras are one of the most inexpensive and readily available automatic sensors and the quantity of information they provide is enormous.

Sight empowers humans to perform many tasks which do not require higher thinking, which are the type of tasks we commonly wish to automate. Therefore automatic vision systems have widespread applicability in, for example, industrial automation. Yet even after many years of research vision systems for high-level vision tasks such as object recognition are very difficult to build. This is true because the appearance of objects follows an extremely complicated physical process.

The key to reliable high-level vision processing is low-level processing able to summarize and extract characteristic and robust features from the raw image data. In fact basic feature detection algorithms such as Canny's edge detector [15] to the more contemporary SIFT operator [37] are among computer vision's most important developments. Low-level processes also exist in the human visual system.

The focus of this thesis is a particular low-level vision technique called **image segmentation**. Image segmentation refers to partitioning a digital image into segments based on pixel information properties, such as colour and proximity. Segmentation is commonly cited as one of the most important preliminary steps of a vision system because it summarizes a large number of pixels into a manageable number of segments, and because segments have important compound information such as shape, texture, and topology which are not available from the raw pixel data. The segmentation data structure can benefit many image processing tasks, for example, graphics with adaptive

level-of-detail or systems for object recognition.

One particularly useful type of segmentation is the hierarchical segmentation [47] which will be the focus of our work. A hierarchical segmentation is a nested partition of image pixels. The recursive data structure can be visualized graphically as a tree. One segment (the root) covers the entire image, which branches into increasingly smaller segments thereby embodying different levels of abstraction. The hierarchical segmentation generalizes the partition segmentation, as we may always reproject a horizontal slice of the hierarchy to obtain a proper partition of pixels. The hierarchical segmentation is particularly useful for analysis and visualization of image data [59].



Figure 1.1: A digital photograph and two renderings of an image segmentation.

To put segmentation into context we may view it as a case of cluster analysis, or simply clustering, on image data. Image segmentation is regarded as ‘unsupervised’ clustering because classes are not known beforehand, and though it is not always interpreted this way (see for example [73]), we apply the constraint that segments (clusters of pixels) should be contiguous. Many of the principles discussed in this thesis also apply to other clustering problems (such as hierarchical clustering).

There exists a large body of work on algorithms for computing image segmentations. There are two main strategic considerations in the design of a segmentation algorithm [64]:

1. What is the precise criterion for a good partition? What function defined on a segmentation increases (or in case of a logical criterion, becomes satisfied) with increasing quality of the segmentation?
2. How do we compute the partition? In other words, what is the algorithm for computing a partition which maximizes (satisfies) the criterion?

In this thesis we investigate Shi and Malik’s first question: What criterion produces a good segmentation? Most generally, in order to facilitate object-based operations on images, the segmentation should be as correct as possible to the organization of physical objects in the image. The purpose of a segmentation criterion is to characterize the visual statistics of distinct objects or other key organizational components of the scene. How do we define a function on raw image data (pixel location and colour) that predicts physical coherence? In other words, what is an object, and what do objects look like?

In Chapter 2, a review of several approaches to segmentation focuses a class called ‘variational criteria’. A variational criterion is a single function that scores the segmentation and normally combines several different measures of the quality of the segmentation. Furthermore, some methods consider the appearance of an image to be the result of a system of underlying random variables. A probabilistic segmentation criterion is specified in the form of assumptions about the feature distribution objects, such as that the colour signature of an object should be compact. In accordance with Bayes’ law we may solve for the **maximum likelihood** segmentation, or if we also assume a prior probability, the **maximum *a posteriori*** segmentation. We build on previous methods hierarchically and argue that the hierarchy is more true to the organization of objects in nature than other models currently used for image segmentation.

Ultimately a segmentation algorithm is only as good as the assumptions made by the criterion about what objects look like. Yet, the segmentation criterion is a function on image appearance only, and not of the physical scene. Without an explicit definition of ‘objects’ to evaluate against we cannot make a quantitative statement about how *correct* any of the hundreds of published segmentation algorithms actually are. The majority of proposed segmentation methods have failed to explicitly address this issue resulting in a lack of rigorous evaluation methodologies in the area. Unnikrishnan, Pantofaru and Hebert [71] observe: “Typically, the effectiveness of a new algorithm is demonstrated only by the presentation of a few segmented images and is otherwise left to subjective evaluation by the reader. Little effort has been spent on the design of perceptually correct measures to compare an automatic segmentation of an image to a set of hand-segmented examples of the same image”. Jiang, Marti, Iniger and Bunke [31] echo: “... segmentation performance evaluation remains subjective. Typically, results on a few images are shown and the authors argue why they look good. The readers frequently do not know whether the results have been opportunistically selected or are typical examples, and how well the demonstrated performance extrapolates to larger sets of images.”

The primary contribution of this thesis is a method for evaluating the segmentation

against a separate ‘ground truth’ data set. We test against the Berkeley Segmentation Data Set (BSDS) presented by Martin, Fowlkes, Tal and Malik [44]. In the Berkeley Segmentation Data Set, human subjects have been shown digital images and asked to:

“Divide each image into pieces, where each piece represents a distinguished thing in the image. It is important that all of the pieces have approximately equal importance. The number of things in each image is up to you. Something between 2 and 20 should be reasonable for any of our images”

Inspecting the ground truth data, one immediate observation is that each subject segments an image differently. Martin *et al.* observe that “even if two observers have exactly the same perceptual organization of an image, they may choose to segment at varying levels of granularity”. Even when the subjects agree on the position, scale and relative importance of objects in the image, the boundary between the same two objects, when juxtaposed, can differ by many pixels in different hand segmentations.

This inspection illustrates why ground truth evaluation of image segmentation is difficult: there does not exist a single ‘correct’ segmentation for natural scenes. For example, we cannot state definitively whether some pair of image pixels belong to the same distinguished thing. Subjects will eventually disagree. But considering this probabilistically, we may at least say that for any pair of image pixels there is a certain probability that the subject will group them together, rather than into different segments.

In an image segmentation we group pixels according to their visual properties, so that pixels in each segment are more visually ‘alike’ than pixels in different segments. We hypothesize that the ground truth state of pixels will also exhibit this grouping property, where the ground truth state of a pixel may be either ‘inside’ or ‘outside’. Empirical sampling shows a strong correlation between the visual statistics, as computed by the proposed segmentation algorithm, and the ground truth data. If a segment contains some ground truth pixels, that they will appear disproportionately in one subsegment or another. This demonstrates that just as pixels within any segment appear ‘alike’, pixels within a segment tend to be jointly ‘inside’ or ‘outside’ a ground truth segment. To put it simply, if a group pixels have similar visual properties then they will often have the same ground truth state.

This is significant because it means we can use a segmentation to predict the distribution of ‘distinguished things’ in an image. To quantify the accuracy of the prediction we measure the **information cross-entropy** of the estimate with respect to the ground truth distribution. **Information entropy** is a measure of disorder or uncertainty in a

probabilistic system. In communication theory, information entropy gives the fundamental limit to the efficiency with which one can exactly specify the state of a probabilistic system. Higher efficiency can be achieved by using shorter messages for more probable states or events. In fact, the optimal message length in a hypothetical optimal language of b symbols is given by the negative logarithm base b of the probability of occurrence [62]. The cross-entropy

$$H(p, q) = \sum_x p(x) \log_b \left(\frac{1}{q(x)} \right).$$

gives the average description length of a state x which occurs with probability $p(x)$, but for which we have computed an optimal language based on our estimate $q(x)$. For instance a logarithm base two gives the description length in **bits**.

Other measures exist for computing discrepancy between segmentations (i.e., between hand-labeled and computer-generated segmentations), however the benefit of cross-entropy is that the unit of measure relates directly to a conceptual computer vision application. Suppose there is a communications channel between a server and a client, which both have a copy of an image and its segmentation. Now, suppose the server wishes to label or identify a subset of pixels which represent a ‘distinguished thing’. The client is able to specify the ground truth state of pixels more efficiently because correlated pixels have been grouped together. The more closely the segmentation hierarchy matches the ground truth data the more efficient it will be for this task. After optimization our hierarchical model is able to describe ground truth segments with roughly 2 per cent of the description length used in a scheme which disregards pixel groupings.

1.1 Contributions

The primary contribution of this thesis is a method for evaluating the segmentation against a separate ‘ground truth’ data set. We do this by considering the data set to be, and by estimating, a ground truth probability distribution. This is the joint distribution of pixel ground truth state for a single ground truth segment. The error of the estimate is given by the cross-entropy between it and the actual ground truth. The result is an evaluation procedure that is task-based (allowing clear interpretation of the results) and general (allowing to compare results laterally across experimental conditions, even different models).

We give a hierarchical, probabilistic model for the visual appearance of an image. Our intent is to generalize existing models into a form more suitable for the type of

images in the BSDS (see Figures 6.4-6.7). We argue that the hierarchical dependency structure is better suited for general photography (at least, the images in the BSDS database) than existing probabilistic models. We define a segmentation criterion in and implement the region merging segmentation algorithm. We are not aware of any other work using a region merging algorithm to solve for a completely hierarchical, coordinate independent, generative model of image appearance. Our use of the beta distribution to model segment size is unique.

We give a hierarchical, probabilistic model for the ground truth state of image pixels. This model is based on a novel approximation for the expected number of ground truth pixels in subsegments, given the number of ground truth pixels in a segment.

We give an efficient algorithm, the segment description algorithm, for computing the segmentation error. Let K be the number of ground truth segments and let N be the number of image pixels. The complexity of the segment description algorithm is $O(KN)$, where the complexity of other error measures can be $O(N^2)$ [72] or $O(N^4)$ [41].

In the experimentation portion of this thesis we explore the real-valued, multi-dimensional space of the parameters of the criterion using an evolutionary optimization method. The optimization system automatically adjusts the parameters of the criterion for best results.

A computer program was developed using the optimized segmentation criterion for the conversion of bitmap images into SVG format vector graphics. This program generates coloured polygons that are expected to be correct with respect to the objects in the image, not just the image appearance. A novel method was also developed for encoding and interactively adjusting the level of detail based on the hierarchy of segments.

A computer program was developed using the optimized segmentation criterion for the separation of the subject from the background of an image. The user labels foreground pixels by selecting image segments in a coarse-to-fine strategy. The application effectively demonstrates the segment description paradigm used for evaluation: the better the algorithm is for segmenting the image into subject and background, the fewer selections will have to be made by the user.

1.2 Summary

In Chapter 2 we review the current state of the art of variational and probabilistic segmentation criteria. The probabilistic graphical models of image formation reviewed in this chapter will provide a grammar for discussing segmentation criteria as well as the proposed probabilistic model for ground truth.

In Chapter 3 we propose a segmentation criterion based on a hierarchical model. This criterion combines the visual cues of size, colour, luminance and shape. We also give a greedy algorithm for computing a segmentation.

In Chapter 4 we review existing methods for the evaluation of image segmentations. Various methods for computing the discrepancy between a segmentation and a ground truth or peer segmentation are given in Section 4.2.

Section 5.1 motivates an information theoretic, ground truth based evaluation. This method is based on the ground truth probability distribution: the distribution of set of human-delineated ground truth segments. Section 5.2 gives a formula for estimating this distribution based on the hierarchical image segmentation. An efficient algorithm for computing the discrepancy between the estimate and the ground truth segments in the data set is given in Section 5.3.

Parameters of the model are optimized experimentally in Chapter 6. Two applications of the optimized segmentation algorithm are given in Chapter 7. Conclusions and future work are discussed in Chapter 8.

Chapter 2

Probabilistic Graphical Models in Image Segmentation

2.1 Variational and Probabilistic Segmentation Criteria

The segmentation criterion of an algorithm is any measurement that is made to ‘score’ a segmentation. Criteria often combine different measures. Ward’s criterion [74] defines a merging predicate which reflects the increase in the sum of squared error which would be introduced by a merge of segments. Texture [17, 20] and colour distributions [41] can also be measured. Van Droogenbroeck and Talbot [20] employ a criterion based on texture, mean difference and size cues. Xuan, Adali and Wang [77] combine the influences of colour, size and connectivity, multiplicatively. Factors may also change at different times over the computation [10, 77]. Brox, Farin and deWith [14] employ three different criteria in different stages: Ward, a combination of Ward and mean difference, and the mean squared difference in pixel intensities along the border. Shu, Bilodeau and Cheriet [65] also employ three stages of criteria: a combination of the Fisher criterion and common boundary length, a combination of segment size and common boundary length, and then common boundary length and segment size in turn. Luo and Guo [38] combine the cues of area, compactness, convexity, colour variance, colour mean, contour continuity and edge strength.

If the cost function, especially a multi-term one, can be expressed as a single expression we can call the method variational. A variational criterion is easier to study compared to, for example, a multi-stage algorithm.

The **Mumford-Shah functional** [49, 48] represents an attempt at a unifying, variational criterion. The score (or ‘energy’, which decreases with score) of a segmentation is

$$E_X(Y) = \omega_1 \sum_i (Y_i - X_i)^2 + \omega_2 \sum_i \nabla \cdot \nabla Y_i + \omega_3 B(Y), \quad (2.1.1)$$

The first term sums the squared error between the grey level of pixel X_i and the grey level predicted by the segmentation Y_i . The second term scores smoothness of the estimate (in case of a piecewise-smooth model), and the last term computes the total length of segment boundaries. Thus, the visual error of the segmentation is weighed, parametrically, against its complexity.

The criterion can also be a probability. In a probabilistic criterion the real world and the imaging process are modeled as the result of an underlying statistical process. In this case a high segmentation score means the segmentation is estimated to be, with high probability, the underlying source of the observed image.

In a probabilistic method developed by Geman and Geman [26], computing the segmentation means computing the maximum *a posteriori* (MAP) estimate of the underlying segmentation Y , that is, the true segmentation. Calculation of the MAP estimate

$$\max_Y P(Y|X) = \max_Y P(X|Y) P(Y) \quad (2.1.2)$$

applies Bayes’ theorem

$$P(X|Y) = \frac{P(Y|X)P(X)}{P(Y)}. \quad (2.1.3)$$

As with the Mumford-Shah functional the complexity of the segmentation $P(Y)$ (the *prior* distribution) is scored against the fidelity to the image $P(X|Y)$ (the *posterior* distribution). In other words, we make assumptions about $P(Y)$ and $P(X|Y)$ and work backwards from the observed image X to obtain the most probable segmentation Y .

LeClerc [36] also presented a Bayesian approach but chose to specify the source image in terms of a descriptive language. In this method the location and mean colour of each segment must be described, followed by the segment boundaries using an edge-chain language, followed by the residual errors of individual pixels. The best segmentation of the image is the one having the simplest description, or **minimum description length**.

Given a probability $P(Y)$, the theoretically optimal length of a description of a random state Y is given by

$$\log \left(\frac{1}{P(Y)} \right). \quad (2.1.4)$$

More likely object appearances are given shorter description lengths for maximal total description efficiency. But,

$$\arg \max_Y P(Y) = \arg \min_Y \log \left(\frac{1}{P(Y)} \right) \quad (2.1.5)$$

“Thus, we see that, by choosing optimal description languages for given prior probabilities, the MDL strategy is equivalent to the MAP strategy. *Conversely*, if one assumes the prior probabilities implicitly specified by the given descriptive languages, the MAP strategy is equivalent to the MDL strategy. The choice of strategies depends on whether it is easier or more natural to specify a descriptive language directly or specify prior probabilities [...] Thus, to model a piecewise constant original image, MAP would specify the appropriate probabilities of each pixel having the same colour as each of its neighbours. Equivalently the length of the description via MDL would measure the length of all boundaries and add the number of segments induced” [36]

Morel and Solimini [48] demonstrated that many existing segmentation criteria including MDL, Ward’s criterion and the normalized cut [64] criterion could be expressed in terms of the same variational criterion, and argued the most general of which is the Mumford-Shah criterion. Similarly, Zhu and Yuille [81] presented a generalization of Bayesian and MDL approaches as well as other methods. These two reports demonstrate that many existing segmentation criteria may be generalized into a single variational criterion.

2.2 Models of Image Formation

The MAP and MDL methods reviewed in Section 2.1 may be called probabilistic criteria. A probabilistic criterion proposes a model, a system of variables and dependencies, underlying the generation of an image. A graph of a probabilistic model can be drawn where small circles indicate random variables of a system and (possibly directed) edges indicate dependencies. Various types of **probabilistic graphical models** are shown in Figure 2.2. Observable variables (coloured in grey) are associated with each pixel, the value of the variable being the colour of the pixel. Hidden variables (coloured in white) represent the state of objects in the scene as well as noise incurred during imaging.

By accurately modeling the probability distribution of observable variables, we may solve for the maximum *a posteriori* value for interior nodes. Accurate modeling of the organization of dependencies in a system is of paramount importance.

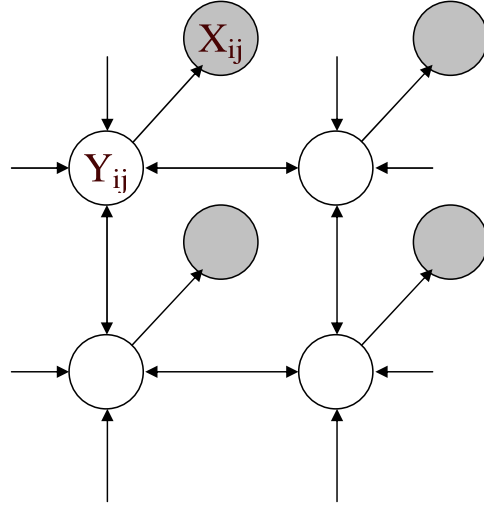
Two ‘probabilistic organizations’ have prevailed in probabilistic segmentation criteria.

Under the **piecewise smooth** model (Figure 2.1(a)), the value of a pixel depends primarily on its immediate neighbours. A pixel in row i and column j is represented by a random variable X_{ij} . A pixel may differ from the unknown ‘true’ appearance at that location Y_{ij} due to noise or damage, but the unknown variables are known to be smooth with respect to their neighbours. Various neighbourhood structures can be used however a grid-like Markov random field closely matches the format of a bitmap image. Some methods [26, 49] augment the structure to model discontinuities in the underlying image with boundary variables, which tend to continue and in straight lines. Local smoothness among adjacent pixels is enforced by dependencies between pixel variables which state that with high likelihood, pixel values are similar to their neighbours, except for when the boundary variable between them is active. This model applies when boundaries are not required to be closed, for example, in edge detection.

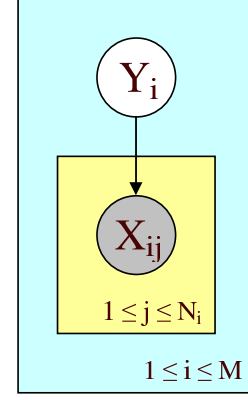
Under the **piecewise constant** model (Figure 2.1(b)), the value of a pixel X_{ij} depends on its segment Y_i , regardless of local proximity within that group. This model assumes that all data can be partitioned into M groups Y_i . The distribution of all N_i pixels in segment i is parameterized by Y_i , explaining why pixels within a segment are more ‘alike’ than pixels in different segments. The analytical benefit to this and other ‘mixture’ models is that the distribution of all data may be decomposed into a set of less complex distributions that are much easier to analyze and estimate.

Unlike the Markov random field, the Bayesian network representing the piecewise constant model is directed. We may think of this model as a top-down selection process for generating data. In Figure 2.1(b), Y_i variables are selected from the distribution $P(Y)$, one for each segment. Next, pixels X_{ij} take values from $P(X|Y)$, thus, they are independent from pixels outside their segment. LeClerc demonstrates the use of this assumption ([36], Section 5). He models the images as a set of piecewise constant regions with additive white noise of known variance. In fact, if noise is assumed Gaussian, then this can be considered a mixture of Gaussians.

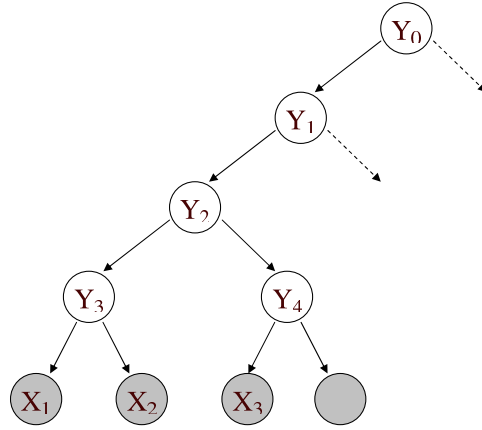
We propose to use a **hierarchical model** that generalizes the mixture model to multiple levels. In the hierarchical generative model, groups of a mixture model are recursively subdivided into smaller mixtures. We may compute a hierarchical segmentation from a hierarchical model by taking a segment for each interior variable.



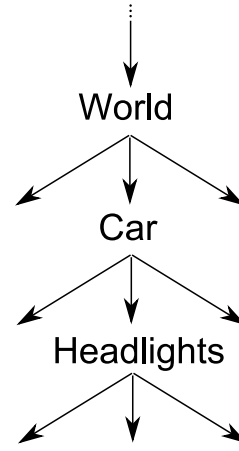
(a) Piecewise Smooth



(b) Piecewise Constant



(c) Hierarchical Model



(d) Object Tree

Figure 2.1: Probabilistic graphical models of image formation. Nodes indicate random variables and arcs indicate statistical dependence. Pixel colours are measured as the value of grey nodes. White nodes are hidden. In the piecewise smooth model, pixel X_{ij} is close in colour to Y_{ij} , which is close in colour to its neighbours (Figure 2.1(a)). In the piecewise constant model a pixel may belong to a finite number of classes. Given class Y_i , the appearance of pixel X_{ij} follows a (possibly Gaussian) distribution. The rectangular container is a ‘plate’ which indicates replication according to the subscript. (Figure 2.1(b)). By stacking several mixture models we obtain a hierarchical segmentation able to reflect the compositional structure of natural objects (Figure 2.1(d))

The proposed hierarchical model is a top-down, generative model of image formation. We observe pixels which are a set of random variables $\{X_1, X_2, \dots, X_N\}$. The model also contains hidden variables $Y = \{Y_1, Y_2, \dots, Y_M\}$. Dependencies follow a connected, directed tree graph where leaves represent pixels variables and internal branching nodes represent hidden variables (Figure 2.1(c)). In the present work we also assume a hidden variable has exactly two ‘children’, which can be pixel variables or other hidden variables. The set of pixels descendent from internal variable Y forms an image segment.

In the hierarchical model,

$$P(X_1, X_2, \dots, X_N, Y_1, Y_2, \dots, Y_M) = P(Y_{root}) \prod_{i=0}^M P(Children(Y_i)|Y_i). \quad (2.2.1)$$

To illustrate, the approach considers the creation of an image as a top-down selection process on a tree of objects. A root object has appearance Y_{root} with probability $P(Y_{root})$. Objects S are then recursively divided into subobjects having appearance L and R with probability $P(L, R|S)$. That is, the states of child segments R and L are chosen randomly based on the state of the parent segment S .

The mixture function $P(Children(Y_i)|Y_i)$ explains the observation that pixels can be subdivided into groups of ‘alike’ pixels, at all levels of scale. That is, the hierarchical model is able to relax the constraint of a partition model that data either related (if they are in the same segment) or unrelated (in different segments). In Figure 2.1(c), pixels X_1 and X_2 are more closely related statistically than pixels X_1 and X_3 . This model arises from two simple observations which we may make about objects in natural images. First, objects exist in different scales in a scene. The statement that a car headlight is an object is no less valid than the statement that a car is an object. Large-scale objects are commonly compositions of small-scale objects. A car is a good example of an object that is composed of many smaller objects. The multi-level segmentation allows to directly represent the multi-level nature of objects. Second, the transition between two objects may be gradual, in which case it is impossible to divide the objects exactly. While the piecewise-smooth model can model gradual changes, and the hierarchical model cannot, the hierarchical model can still model the composition relationships above and below a gradual change. That is, it can capture spatial non-uniformity to a certain extent.

Martin *et al.* [44] observe about their own data “one can think of a human’s perceptual organization as imposing a hierarchical tree structure on the image.” Unlike the piecewise constant model, the hierarchical model allows pixels to be statistically dependent to various degrees. Compared to the piecewise smooth model the internal tree structure

of the hierarchical model offers a better internal structure for object-based visualization and analysis of the image.

Hierarchical models exist in multiscale image analysis (See Choi and Baraniuk [17], Vincken, Koster and Viergever [73]) and scene understanding (See Parikh and Chen [53]) but have received less attention in hierarchical image segmentation. Recent developments in mutual information based clustering have involved hierarchical models [33, 2].

Chapter 3

A Hierarchical Image Segmentation Algorithm

3.1 A Hierarchical Segmentation Criterion

We propose a probabilistic segmentation criterion based on the hierarchical model (Figure 2.1(c)). Let segmentation S be a set of three-tuples of segments $\{S, L, R\}$, such that R and L are the left and right children of S in a hierarchical model. Let $X := \{X_i\}, 1 \leq i \leq N$ be the ‘leaf’ variables and let $Y := \{Y_i\}, 1 \leq i \leq M$ be the latent variables. Let \mathbb{S} be the space of visual feature measurements of an image segments. There are four visual properties of an image segment we wish to model: size, colour, luminance and shape. Therefore there are four random elements for any segment Y :

$$Y := \{Y_{\text{size}}, Y_{\text{colour}}, Y_{\text{luminance}}, Y_{\text{shape}}\}. \quad (3.1.1)$$

Let segmentation S be a set of three-tuples of segments $\{S, L, R\}$, such that R and L are the left and right children of S in a hierarchical model. Let us define a distribution

$$q : \mathbb{S}^3, \mathbb{O} \rightarrow \mathbb{R}$$

for the conditional distribution of child segments on a parent segment. That is,

$$\begin{aligned} q(S, L, R, \Omega) \\ \text{is an estimate of} \\ P(L, R|S). \end{aligned} \quad (3.1.2)$$

Also, let Ω be a random variable in \mathbb{O} , the space of fixed parameters of the system. Therefore,

$$\begin{aligned}
 P(Y_0) \prod_{\{S,L,R\} \in S} q(S, L, R, \Omega) \\
 \text{is an estimate of} \\
 = P(Y_0) \prod_{\{S,L,R\} \in S} P(L, R|S) \\
 = P(Y_0, Y_1, \dots, Y_n, X_0, X_1, \dots, X_n)
 \end{aligned} \tag{3.1.3}$$

where $P(Y_0)$ is the prior on the root segment which we take to be constant.

Distribution q is also defined in four parts:

$$\begin{aligned}
 q(S, L, R, \Omega) &:= q_{\text{size}}(S, L, R, \omega_{\text{size}}) \\
 &\quad \times q_{\text{colour}}(S, L, R, \omega_{\text{colour}}) \\
 &\quad \times q_{\text{luminance}}(S, L, R, \omega_{\text{luminance}}) \\
 &\quad \times q_{\text{shape}}(S, L, R, \omega_{\text{shape}})
 \end{aligned} \tag{3.1.4}$$

This distribution is defined in the following sections based on what we know about objects, or at least, desirable segments. A higher value indicates a ‘more probable’ appearance segment. The general approach is that subsegments, or parts, of objects tend to have similar visual qualities. We measure a segment’s size S_{size} , its mean colour on the a*b* colour axis S_{colour} , its mean luminance $L^* S_{\text{luminance}}$, and its boundary length S_{shape} . Variable S_{size} refers to the number of pixels in the segment. In this implementation of the hierarchical model we assume a segment has exactly two child segments, unless the segment has only one pixel, in which case it is a leaf.

We hypothesize that if q is a good estimate of the appearance of real world objects then the computed segmentation will be also be close in structure to hierarchical objects in the real world.

3.1.1 A Hierarchical Model for Segment Size

The size of a segment is an important indicator of its visual importance in the scene. If a segment is divided into two subsegments of equal importance, then we may expect that they will have roughly the same size. In their database of human delineated segments, Martin *et al* [44] observe an exponential fall-off in the frequency of larger segments. They fit a curve of the form

$$y = Ax^\alpha \tag{3.1.5}$$

where $\alpha = -1.008$ and x is region area. However, this formula does not produce the desired behaviour when conditioned on exactly two subsegments. This curve is strictly decreasing for $\alpha < 0$, constant at $\alpha = 0$, and strictly increasing at $\alpha > 0$ and cannot account for a curve that is modal at $x = \frac{S_{\text{size}}}{2}$.

We suggest instead to fit a beta distribution over the relative sizes of L and R :

$$B(x, \alpha, \beta) = \frac{1}{B(\alpha, \beta)} x^{\alpha-1} (1-x)^{\beta-1} \quad (3.1.6)$$

which is the distribution of the $(\alpha+1)$ th highest value of $\alpha+\beta+1$ samples of the uniform distribution over $[0, 1]$ and $B(\alpha, \beta)$ is the normalizing beta function.

Given that $L_{\text{size}} + R_{\text{size}} = S_{\text{size}}$, we model the probability density or relative segment size size using the Beta distribution (Figure 3.1)

$$q_{\text{size}}(S, R, L, \omega_{\text{size}}) = c \left(\frac{L_{\text{size}}}{S_{\text{size}}} \right)^{\omega_{\text{size}}-1} \left(\frac{R_{\text{size}}}{S_{\text{size}}} \right)^{\omega_{\text{size}}-1}, \quad (3.1.7)$$

where c is a constant. For

$$\omega_{\text{size}} = \alpha = \beta,$$

the expected value will be at

$$\mathbb{E} \left(\frac{L_{\text{size}}}{S_{\text{size}}} \right) = \frac{\alpha}{\alpha + \beta} = \frac{1}{2}.$$

Also, for

$$\omega_{\text{size}} = \alpha = \beta > 1.0,$$

the distribution will be modal at

$$\frac{L_{\text{size}}}{S_{\text{size}}} = \frac{R_{\text{size}}}{S_{\text{size}}} = \frac{1}{2}.$$

The multidimensional generalization of the beta distribution is the **Dirichlet distribution**. The Dirichlet distribution has been used previously as a size prior in mixture models [12, 67]. It has not yet been shown whether the size of BSDS ground truth segments follow a Dirichlet distribution. Our observations would support this, since the marginal of a Dirichlet down to a single variable is the beta distribution.

3.1.2 A Hierarchical Model for Segment Mean Luminance

We expect that the luminance values for pixels within a segment to be ‘alike’. Therefore, we model the difference between a the mean luminance of a parent S and child R as a

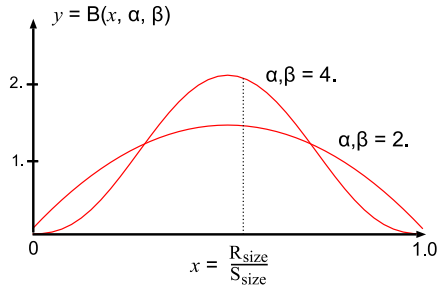


Figure 3.1: The beta distribution as the distribution of R_{size} given S_{size} .

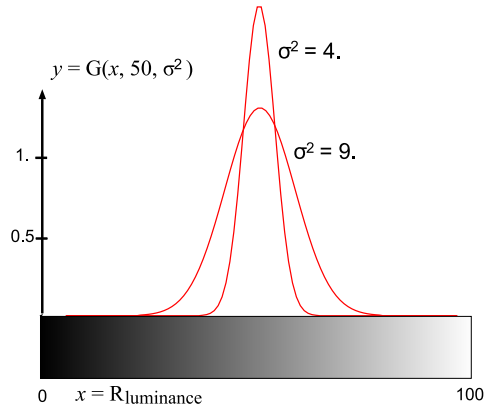


Figure 3.2: The Gaussian distribution. The distribution of $R_{\text{luminance}}$ given $S_{\text{luminance}} = 50$

Gaussian-distributed vector x

$$G(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(\mu - x)^2}{2\sigma^2}\right) \quad (3.1.8)$$

centered at $S_{\text{luminance}}$ (Figure 3.2).

For children L and R of S ,

$$\begin{aligned} q_{\text{luminance}}(S, L, R, \omega_{\text{luminance}}) \\ &= G\left(L_{\text{luminance}}, S_{\text{luminance}}, \frac{S_{\text{size}}}{L_{\text{size}}} \omega_L^2\right) G\left(R_{\text{luminance}}, S_{\text{luminance}}, \frac{S_{\text{size}}}{R_{\text{size}}} \omega_L^2\right) \\ &= c \exp\left(\frac{(L_{\text{luminance}} - S_{\text{luminance}})^2 L_{\text{size}}}{S_{\text{size}}}\right) \exp\left(\frac{(R_{\text{luminance}} - S_{\text{luminance}})^2 R_{\text{size}}}{S_{\text{size}}}\right) \end{aligned} \quad (3.1.9)$$

where c is a constant.

We have decided to scale the variance according to segment size after observing that small segments with outlying values contributed too much to the merge score.

3.1.3 A Hierarchical Model for Segment Mean Colour

The derivation of colour is the same as that for luminance, except that S_{colour} is the mean in the a^*b^* colour axes, rather than the L^* axis, and a different parameter is used.

For children L and R of S ,

$$\begin{aligned} q_{\text{luminance}}(S, L, R, \omega_{\text{colour}}) \\ &= G\left(L_{\text{colour}}, S_{\text{colour}}, \frac{S_{\text{size}}}{L_{\text{size}}} \omega_L^2\right) G\left(R_{\text{colour}}, S_{\text{colour}}, \frac{S_{\text{size}}}{R_{\text{size}}} \omega_L^2\right) \\ &= c \exp\left(\frac{(L_{\text{colour}} - S_{\text{colour}})^2 L_{\text{size}}}{S_{\text{size}}}\right) \exp\left(\frac{(R_{\text{colour}} - S_{\text{colour}})^2 R_{\text{size}}}{S_{\text{size}}}\right). \end{aligned} \quad (3.1.10)$$

3.1.4 A Descriptive Language for Segment Perimeter

In the case of of segment shape, we assume that objects tend to be spatially compact. The shape term of the criterion increases with segment perimeter, thereby biasing convex shapes.

Instead of a probability we define the shape criterion in terms of a proper description language. Following the method of LeClerc [36], we describe the shape of subsegments

by first addressing a single pixel, from which a chain-code begins that traces a path until a partition is made. Let $B(S)$ is the boundary length of segment S , or more precisely, it is the length of the message required to describe the boundary of S using a language with b symbols. Also let C be the length required to address the first element of the chain. The the description length of a subdivision is given by

$$C + \frac{1}{2 \log(b)} \text{CB}(R, L) = C + 2\omega_{\text{shape}}(R_{\text{shape}} + L_{\text{shape}} - S_{\text{shape}}) \quad (3.1.11)$$

where $\text{CB}(R, L)$ is the length of common boundary between the two.

Let us assume the prior

$$-\log(P(S_{\text{shape}})) = C + 2\omega_{\text{shape}}S_{\text{shape}}. \quad (3.1.12)$$

Therefore, the size term of the criterion is

$$-\log(q_{\text{shape}}(S, R, L, \omega_{\text{shape}})) = C + \omega_{\text{shape}}(R_{\text{shape}} + L_{\text{shape}} + S_{\text{shape}}) \quad (3.1.13)$$

3.2 Computing the Segmentation

We solve for the segmentation hierarchy with the region merging algorithm [41]. The standard implementation begins with each pixel in its own segment and iteratively merges neighbouring segments until only one segment remains. Each iteration performs the merge which achieves stepwise optimization of the criterion [8]. For this reason the segmentation criterion is sometimes called the merging criterion. In the context of the proposed model each iteration adds a segment to the model that is most probable given the existing segments. Segmentation S (Algorithm 1 defines a ‘forest’ of hierarchical models until the final merge completes the tree structure (Figure 3.3). This algorithm is greedy and is not guaranteed to find the optimal segmentation. We employ region merging because it is straightforward, tractable, allows diverse segmentation criteria to be used interchangeably, and because it implicitly builds a hierarchical segmentation. It also allows a probabilistic criterion: maximization of the criterion reflects the maximum *a posteriori* estimate given the observed image, computed as follows:

$$\max_Y P(Y_0, Y_1, \dots, Y_n | X_0, X_1, \dots, X_n) = \max_Y P(Y_0, Y_1, \dots, Y_n, X_0, X_1, \dots, X_n) \quad (3.2.1)$$

where $Y = \{Y_0, Y_1, \dots, Y_n\}$ are hidden variables and $\{X_0, X_1, \dots, X_n\}$ are one-pixel ‘leaf’ variable initialized directly from the bitmap.

Algorithm 1 (Region Merging Algorithm)

1. Initialize a segment X for each pixel in the image. Let G^0 be initially the set of all segments and let segmentation S be empty. Initialize $t = 0$.
2. At each iteration, select segments R and L from G that are most likely to be siblings,

$$\{\hat{R}^t, \hat{L}^t\} = \arg \max_{R, L \in G^t} q(S, L, R, \Omega) \quad (3.2.2)$$

where

$$\hat{S}^t = \arg \max_S q(S, \hat{L}^t, \hat{R}^t) \quad (3.2.3)$$

and merge them

$$\begin{aligned} G^{t+1} &= G^t \cup \hat{S}^t \setminus \{\hat{L}^t, \hat{R}^t\} \\ S &= S \cup \{\hat{S}^t, \hat{L}^t, \hat{R}^t\} \\ t &= t + 1. \end{aligned} \quad (3.2.4)$$

3. End when G^t contains only one segment; the root of the hierarchy.





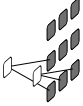

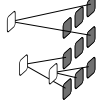

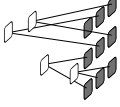

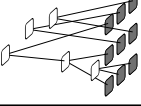

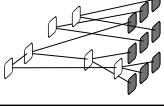

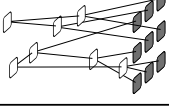

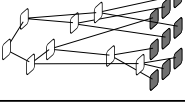
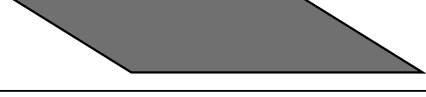
t	S	G^t
1		
2		
3		
4		
5		
6		
7		
8		
9		

Figure 3.3: The region merging algorithm. Each iteration, the two most similar adjacent segments in G^t are merged. This adds to segmentation S a parent segment which achieves stepwise optimization of the criterion q . Under the hierarchical model, the joint probability distribution of the nine pixel states is given by the product of eight factors (Equation 3.1.3).

Chapter 4

Related Work on Methods for Segmentation Evaluation

Quantification of results is an important step in scientific method, yet despite the large number of proposed and available segmentation algorithms there is a relative lack of performance quantification in the area. Unnikrishnan, Pantofaru and Hebert [71] observe: “Typically, the effectiveness of a new algorithm is demonstrated only by the presentation of a few segmented images and is otherwise left to subjective evaluation by the reader. Little effort has been spent on the design of perceptually correct measures to compare an automatic segmentation of an image to a set of hand-segmented examples of the same image”. Jiang, Marti, Iniger and Bunke [31] echo: “... segmentation performance evaluation remains subjective. Typically, results on a few images are shown and the authors argue why they look good. The readers frequently do not know whether the results have been opportunistically selected or are typical examples, and how well the demonstrated performance extrapolates to larger sets of images.”

Palmer, Dabis and Kittler [52] also note that one of the most important benefits of evaluation is so that algorithms may be self-optimizing. This is particularly applicable for algorithms having one or more parameters.

This chapter will review methods for the evaluation of image segmentations and segmentation algorithms. In Section 4.1 we discuss a different approaches to segmentation evaluation and in Section 4.2 we review various methods for computing the discrepancy between segmentations, a critical yet unsolved problem in ground-truth based segmentation evaluation.

4.1 Approaches to Segmentation Evaluation

Jiang *et al.* [31] give the following taxonomy of ‘the various methods of performance evaluation’:

1. theoretical evaluation
2. experimental evaluation
 - (a) feature-based evaluation
 - i. non-GT (ground truth)-based evaluation
 - ii. GT-based evaluation
 - (b) task-based evaluation

A theoretical evaluation of an algorithm may be performed before it is even implemented. This includes the type of input required, the type of output produced, as well as the asymptotic and expected running time and storage requirements. Algorithms may vary by whether they process greyscale or colour imagery. This distinction was important in the early 1980s when colour digital imagery became more prevalent. Of course, there also other types of images that we may wish to segment. A LANDSAT satellite photo has 7 colour bands. A Magnetic Resonance Image (MRI) does not record colour at all, but rather density.

Some segmentation algorithms are designed to output a partition segmentation rather than a hierarchical one. Also, both of these differ in whether they can divide the image into many segments or only two at a time (thresholding and early snake implementations only produce bipartitions).

Another theoretical topic is the benefit to having fewer parameters. From a theoretical viewpoint being non-parametric suggests generality of the approach and a solid foundation of the problem at hand. In practical use fewer parameters affords lesser ‘fine tuning’ of the algorithm. In either case being non-parametric does not always mean the algorithm will perform better in practice.

In a theoretical evaluation of a segmentation algorithm, Tao and Crisp [69] obtain a theoretical performance bound for the region merging algorithm (Algorithm 1).

Non-ground truth based evaluation methods are measures of some desirable statistic related only to the image and the segmentation itself, not using any ground truth data. The evaluation statistic can be the segmentation criterion itself.

An example of this type of test would be to implement several approximation methods which seek to maximize the same criterion on the same set of images and observe which produces the better final values. This would be the evaluation paradigm we would use if we were examining Shi and Malik’s second question [64] (how to best compute the segmentation).

Ground truth based methods are dominant in current research. Zhang [79] calls ground-truth based methods ‘empirical discrepancy’. Empirical discrepancy methods measure how well the segmentation results match some previously specified ground truth. As these terms suggest two items are required for this test: a ground truth data set labelling the true objects in the image, and a method for computing the discrepancy between the segmentation and the ground truth. Discrepancy can be any measure of similarity or dissimilarity between the automatic segmentation and the goal.

Using a synthetic image it is possible to compare the results of the segmentation to the truth directly. In this case we assume that the synthetic images are a good enough approximation of the type of images that the algorithm will encounter during its working life. The alternative is to hand-label natural images. The Berkeley Segmentation Dataset (BSDS) presented by Martin, Fowlkes, Tal and Malik [44] provides a database of images that have each been partitioned according to the intuition of human subjects, and is as close as we have come to a standard test for segmentation algorithms. Comparing against this data, however, is difficult because “there is no single ground truth segmentation against which the output of an algorithm may be compared [71]”. In other words, each subject segments images differently.

Choice of discrepancy measure is also a matter for debate. Section 4.2 reviews methods for partition and hierarchical discrepancy, including two measures proposed by Martin *et al.* [44] and two other measures which have since been proposed for the same data set [72, 16].

In task-based evaluation the segmentation is performed as a pre-processing step for some higher-level process, for which there is some other measure of performance. Zhang and Gerbrands [80] call this the ultimate measurement accuracy because if the higher-level performance measure is exact then there can be no better quality certification for the segmentation algorithm. Borra and Sarkar [13, 43] argue that “segmentation or grouping performance can be evaluated *only* in the context of a task”. This position is also taken by Palmer, Dabis and Kittler [52], who study evaluation of the related problem of boundary detection.

Section 5.1 will present an evaluation based on the task of ‘describing’ ground truth segments.

4.2 Measures of Discrepancy for Empirical Evaluation

A central task in many approaches to evaluating image segmentations is the comparison of a segmentation to either ground truth or a peer segmentation. Most methods operate on two partitions. We will review these and some other methods that have been used in the larger context of hierarchical clustering.

The comparison of clusterings for the purpose of cluster validation has been studied in various capacities over the years. For example, in the early 1980s the topic was intensely studied by biologists who desired to validate and compare genetic dendrograms (tree graphs of hierarchical clusterings).

4.2.1 Partition Consensus Methods

There are a great number of partition consensus methods based on a tally of the number of pairs of pixels which appear in the same segment versus pairs appearing in different segments. These tallies (or the relative probabilities) may be represented in the 2 by 2 matching matrix:

$$M = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

where the left and right columns indicate respectively the number of pairs which occur in the same and in different segments of one partition and the top and bottom rows indicate respectively the number of pairs which occur in the same and in different segments of the other partition. In general, we wish to observe a correlation among these events: if two pixels appear in the same segment in one partition, then they should also in the other partition. The correlation of the matrix indicates the extent of agreement between the partitions and has been computed in a number of ways.

The Rand index [55] is given by:

$$C_R = \frac{a + d}{a + b + c + d}.$$

Variations were later proposed. Probabilistic Rand [70] allows comparison against a suite of ground truths. The Adjusted Rand [29] and Normalized Probabilistic Rand [71]

indices correct for chance by considering the expected value of the index. This allows index results to be averaged and compared across images.

Other measures based on the matching matrix are the Jaccard index [9]:

$$C_J = \frac{a}{a + b + c}$$

And the Fowlkes and Mallows index [24]:

$$C_{FM} = \frac{a}{\sqrt{(a + b)(a + c)}}.$$

The authors of the BSDS introduced a pair of measures, the global consistency error (GCE) and local consistency error (LCE) [44]. These measures focus on allowing one directional refinements. Suppose segment S_1 contains pixel x in one segmentation and segment S_2 contains the pixel in the other. Then, the directional local refinement error is defined as

$$C_{CE} = \frac{|S_1 \setminus S_2|}{|S_1|}$$

where \setminus denotes set difference and $||$ denotes set cardinality. Errors fall in the range $[0, 1]$ and are averaged over all pixels. “As observed by the authors [,] there are two segmentations that give zero error for GCE and LCE - one pixel per segment, and one segment for the whole image. This adversely limits the use of the error functions to comparing segmentations that have similar cardinality of labels [71]”.

The method of Cardoso and Corte-Real [16] uses the partition discrepancy. The partition discrepancy is given by the minimum number of pixels which must be removed from the image so that both partitions are completely in agreement.

Meilă [45] proposed the variation of information (VI), a novel consensus method based on information theory. VI measures conditional entropy between clusters of each data set, roughly, how much the knowledge of a pixel’s segment in one segmentation reduces our uncertainty about its segment in another segmentation. Because conditional entropy is not symmetric (see Equation 5.1.5), VI adds the conditional entropy in both directions. VI produces a measure with a tangible unit: the bit. Therefore, the author claims the information based approach has better comparability over experimental conditions, for example, when there are different number of segments in each partition.

Ground truth evaluation methods for edge detection are related [78, 52].

4.2.2 The Consensus of Hierarchies

Among correctness-based evaluation methods there has not, to our knowledge, been one to evaluate hierarchical segmentations. In this section we shall review methods which have been used to compare hierarchies and trees in related problem domains.

An early method proposed by Fowlkes and Mallows [24] extends a pairs-counting method by ‘cutting’ two hierarchies (horizontally) each at level k and computing the correlation of the matching matrix. Correlation is then plotted against k . See [24] for related methods.

One statistic on hierarchies of populations that analysts consider is the ‘rank’ of pairs of samples. The rank of a pair of pixels which is the lowest level at which the samples are joined in the same group in the hierarchy (analogously to degree of relation in a family tree). This information can be represented in a triangular matrix, where element $A_{i,j}$ is the (possibly normalized) level at which samples i and j are first joined into the same cluster. The rank gives an indication of how closely related the two samples are, and can be computed in log-linear time. Lapointe and Legendre [34] review two measures of association

$$Co = \sum_i \sum_j \min(A_{i,j}, B_{i,j})$$

and

$$Cu = \sum_i \sum_j |A_{i,j} - B_{i,j}|$$

which are called, respectively, the organized and unorganized complexity by Day [18], and the consensus similarity and metric dissimilarity by Faith and Belbin [21]. Other rank methods include the cophenetic correlation coefficient [66] and rank correlation [32]. The difference between partition (hierarchy) rank and property (distance in sample space) rank has previously been used as a cluster validation indicator [6]. Motivated by work on stability of hierarchical clustering [5], our earlier work used a rank method to compute stability between segmentations under small changes to camera angle [41].

More recently, work has been done to formulate statistical significance tests for these measures [35, 63, 21]. This involves formulating a null hypothesis: the distribution of the consensus index for random hierarchies. As can be imagined, these studies vary in their representation and interpretation of hierarchies and their consensus.

Tree-edit distance, based on topological *operations*, or elementary transformations, of trees, came to the fore particularly in the field of theoretical computer science, where a tree represents a program or data structure rather than a nested clustering. Such methods

include the nearest neighbour interchange of Waterman and Smith [75] for unrooted trees and the metric of Robinson and Foulds [56] for two unrooted trees of arbitrary degree. “Unfortunately this kind of distance [...] may be very hard to compute effectively [7]”.

Consensus tree methods are more focused on the topology of the hierarchy. A consensus tree is a dendrogram which represents information common to one or more rival trees. The exact formulation of the consensus tree depends on what information contained in the rivals are considered important.

In support of consensus tree methods, Mickevich [46] offers an example comprising two similarity matrices for a set of the same four taxa (data to be clustered, also, leaves in the dendrogram). In the first, taxa a and b form a tighter cluster and than do taxa c and d . In the second, taxa c and d are more closely associated than are a and b . In both cases, using nested set notation, the proper hierarchical clustering is $\{\{a,b\},\{c,d\}\}$. Yet, the cophenetic correlation coefficient indicates that the two matrices are negatively correlated.

The Adams-2 [30] consensus tree is somewhat of a cross-product between hierarchical partitions (trees having unlabeled interior nodes). Thus the recursive formula for computing the Adams-2 consensus between two hierarchical partitions of a set of is as follows:

Algorithm 2 (Adams-2 Consensus Trees)

Initialize a set of all taxa.

1. *If the set contains only one leaf, the consensus is the set containing that leaf.*
2. *Otherwise, find in both trees the smallest sets containing the subset in question. Compute the cross-partition. Repeat for each of the smaller sets.*

Alternatively the strict consensus tree contains only the subsets which appear exactly in both trees. Strict consensus trees are also known as majority-rule consensus n-trees [42], also the cladogram of replicated components [51]. Wilkinson [76] proposed the reduced Adams consensus and reduced cladistic consensus trees to address the problems of sensitivity and ambiguity in strict and Adams-2 consensus trees, respectively. A related concept to consensus trees are maximally parsimonious trees [46].

Consensus trees give a representation of the agreement between dendrograms but do not provide a numerical index upon which we can summarize that one pair are more or less in agreement than another pair. For this we may apply indices of tree size or ‘information content’ to the consensus tree. As an example of such an index, the Nelson

and Platnick [51] total information can be computed by iterating over all leaves and summing the number of ancestors. Mickevich [46] applied Farris' distortion measure [22] to Adams-2 consensus tree. Schuh and Polhemus [61], based on the work of Nelson [50], count the number of subsets (component information) in the strict consensus tree. Day [19] elaborates and designs a dissimilarity measure computing the number of informative components in rival trees T_1 and T_2 but not in their strict consensus, $C(T_1, T_2)$. He denotes the cluster representation of a tree T by T' :

$$D(T_1, T_2) = |T'_1| + |T'_2| - 2|C(T_1, T_2)'|$$

In many methods it seems consensus trees (CT) and consensus indices (CI) have been combined nondiscriminately [60], even though “different criteria embodied in the diverse CT and CI methods will lead to different conclusions concerning the same pairs of trees [63]”.

Chapter 5

Segment Description: An Efficient Task Based Segmentation Evaluation

5.1 An Information Theoretic Measure for Discrepancy

While we deliberate greatly over the proper measure of consensus we feel the Berkeley Segmentation Dataset [44] (BSDS) ground truth is, at least, as good as any we would be able to acquire.

In our experiments, a **ground truth segment** is a subset of pixels that have been selected from an image according to some common property. Let us denote a ground truth segment with a vector $\mathbf{v} := [\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3, \dots, \mathbf{v}_N]$ of values in $\{0, 1\}$ where $\mathbf{v}_i = 1$ indicates the i th pixel is ‘inside’ the ground truth segment and $\mathbf{v}_i = 0$ indicates otherwise. Like an image segment, a ground truth segment must be connected.

The ground truth segments in the Berkeley Segmentation Dataset (see Figures 6.4-6.7) correspond to ‘distinguished things’ in the image. Although the BSDS is given in the form of partition segmentations, we consider the data set as a ‘bag’ of segments $V := \{\mathbf{v}^i\}, 1 \leq i \leq K$. K is the number of ground truth segments for all subjects, given a specific image.

Suppose there exists a distribution over variables $X := \{X_i\}, 1 \leq i \leq N$

$$P(X_1, X_2, X_3, \dots, X_N), \quad (5.1.1)$$

which is the true but unknown probability, given an image, that for a randomly selected ground truth segment \mathbf{v} , $\mathbf{v}_i = X_i$, for all $1 \leq i \leq N$. P , the ground truth segment

distribution, models the spatial extent of a single randomly chosen ground truth segment from a randomly chosen subject. That is, it is the probability that a human observer will select \mathbf{v} as a ‘distinguished thing’ given a specific image.

The marginal of this function down to one dimension

$$P(X_i) = \sum_{\mathbf{v}_1=0,1} \sum_{\mathbf{v}_2=0,1} \sum_{\mathbf{v}_3=0,1} \dots \sum_{\mathbf{v}_N=0,1} P(\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3, \dots, X_i, \dots, \mathbf{v}_N) \quad (5.1.2)$$

is equal for all i variables, because each pixel appears in the same number of ground truth segments.

The marginal of this function down to two dimensions

$$P(X_i, X_j) = \sum_{\mathbf{v}_1=0,1} \sum_{\mathbf{v}_2=0,1} \sum_{\mathbf{v}_3=0,1} \dots \sum_{\mathbf{v}_N=0,1} P(\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3, \dots, X_i, \dots, X_j, \dots, \mathbf{v}_N) \quad (5.1.3)$$

is the probability of two pixels being inside a randomly-chosen ground truth segment. This all-pairs model of ground truth is used in the (Normalized) Probabilistic Rand by Unnikrishnan, Pantofaru and Hebert [72, 70].

We are not aware of any evaluation methods based on the full, N -dimensional ground truth probability distribution. Yet, we feel that this is the proper quantity to evaluate against because it most clearly characterizes the ground truth data, that is, the distribution of ‘distinguished things’ in the image.

Now, just as we have proposed a hierarchical model for joint probability of pixel appearance in Section 3, we also propose a hierarchical model for the joint probability of pixel ground truth state. Section 5.2 presents a hierarchical model for ground truth based directly on an estimate produced by the segmentation. Since the visual data and ground truth data are presumably influenced by the same physical influences we should observe a correlation. That is, if two pixels have similar visual properties then they will also often have the same ground truth state.

It has been noted about the BSDS and similar images that distinguished things tend to be refinements of other distinguished things. Ground truth segments tend to include or be included in other segments, but not otherwise overlap, which is a property of a nested partition. This suggests strongly that the hierarchical model is appropriate for this distribution.

Considering the ground truth data as well as the segmentation as a distribution of ground truth segments enables the use of existing measures of discrepancy from the study of communication theory. Information entropy [62] is a measure for expressing

uncertainty about a probabilistic system. The information entropy of distribution $P(X)$ is given by:

$$H(P(X)) = - \sum_i P(X = x_i) \log_b (P(X = x_i)), \quad (5.1.4)$$

where i is integrated over all possible states of X . Suppose we wanted to communicate or store the exact state of a variable X . In a hypothetical descriptive language of b symbols, the optimal length of a description for state X is

$$\log_b \frac{1}{P(X)}.$$

A description is manifested as a string of symbols that uniquely identifies the state of a random variable X . More likely states are given shorter description lengths for maximal total description efficiency. Entropy increases with the number of possible states of the variable X as well as the uniformity of the distribution. The unit of entropy is *symbols*, which are linear units of storage or bandwidth. For a language having only two symbols, the unit is *bits*.

Let $Q(X)$ be a known distribution which estimates $P(X)$. Suppose we precompute a descriptive language based on our estimate, $Q(X)$, such that the length of the description for ground truth segment X is

$$\log_b \frac{1}{Q(X)}.$$

We do not give the descriptive language itself, however, the entropy equations give the average number of symbols used in an optimal descriptive language.

The cross-entropy of two distributions

$$H(P(X), Q(X)) = - \sum_i P(X = x_i) \log_b (Q(X = x_i)) \quad (5.1.5)$$

is the average number of symbols required to completely describe the state X drawn from probability $P(X)$ but for which we have computed a theoretically optimal language based on our estimate $Q(X)$. In case $Q(X)$ is a perfect estimate of $P(X)$ the cross entropy will equal the entropy of $P(X)$. Otherwise more symbols than the optimal amount will be used.

The task of ‘describing’ a randomly chosen ground truth segment \mathbf{v} means uniquely identifying the ground truth state of all N pixels. Suppose our communication medium was limited to two symbols. A naive description method might report each \mathbf{v}_i , in order, verbatim. This would require a description length of N bits for each ground truth

segment. Observing that ground truth segments have an average size of $\frac{N}{20}$ we might find that the optimal description length for a single non-ground truth pixel is

$$-\log_2(0.95) \approx 0.074 \quad (5.1.6)$$

and that the optimal description length for a single ground truth pixel is

$$-\log_2(0.05) \approx 4.322. \quad (5.1.7)$$

Therefore we may improve the average description length per pixel to

$$-0.95 \log_2(0.95) - 0.05 \log_2(0.05) \approx 0.286. \quad (5.1.8)$$

By considering pixel correlations, and ultimately the full N -dimensional joint probability, we may obtain a still more efficient language.

The process of describing image segments can be thought of as a computer vision application in itself. The purpose of segmentation is to facilitate real world operations on an image. Specifying which pixels are part of a single ‘distinguished thing’ is certainly a real-world operation. A prediction Q that closely matches the ground truth P will make this task more efficient. Cross-entropy is asymmetric, but seems to be the correct approach for measuring the predictive power of an estimate against a ‘gold standard’.

In a related approach, to substantiate earlier claims that maximally parsimonious (MP) trees are “the most efficient summary of [the taxonomy’s] information content,” An and Sanderson [3] designed particular information coding methods that achieve high compression of character data. They showed that the MP tree was the most efficient under their coding scheme. Also, they found that high compression could still be achieved in the case of conflicting data arising from, for example, “Recombination, lateral gene transfer, hybridization, and other biological processes” which detract from perfect evolutionary organization and make classification difficult.

In this chapter a probability estimate for the distribution of ‘ground truth segments’ is given based on the hierarchical generative model of image formation. By decomposing the ground truth probability function recursively we hope to closely estimate the actual distribution with a minimally complex model.

5.2 A Hierarchical Model for Ground Truth Segments

We propose to estimate the ground truth probability distribution according to the hierarchical model of image formation (Figure 2.1(c)). Let segmentation S be a set of three-tuples of segments $\{S, L, R\}$, such that R and L are the left and right children of S in a hierarchical model. Let $X := \{X_i\}, 1 \leq i \leq N$ be the 'leaf' variables and let $Y := \{Y_i\}, 1 \leq i \leq M$ be the latent variables. Let \mathbb{T} be the space of measurements of 'ground truth' at an image segment. There are two properties of an image segment we wish to model: size and ground truth intersection. Therefore there are two random elements for any segment Y :

$$Y := \{Y_{\text{size}}, Y_{\text{truth}}\}. \quad (5.2.1)$$

Let us define a distribution

$$q_{\text{truth}} : \mathbb{T}^3, \mathbb{O} \rightarrow \mathbb{R}$$

for the conditional distribution ground truth of child segments on a parent segment. That is,

$$\begin{aligned} q_{\text{truth}}(S, L, R, \Omega) \\ \text{is an estimate of} \\ P(L, R|S). \end{aligned} \quad (5.2.2)$$

Also, let Ω be a random variable in \mathbb{O} , the space of fixed parameters of the system. We define function q_{truth} , based on the behaviour we expect the spatial extent of objects to exhibit. Each q_{truth} is a mixture component which explains why we may subdivide pixels into segments such that the pixels within each segment are more 'alike' than pixels in different segments, at all levels of scale.

In this chapter, we consider the size of a segment S_{size} to be known, as this can be calculated given a segmentation. R_{truth} may take an integer value between $LB = \max(0, R_{\text{size}} + R_{\text{truth}} - S_{\text{size}})$ and $UB = \min(R_{\text{size}}, S_{\text{truth}})$ inclusively.

A pixel in S may have one of four states: it may or may not be a ground truth segment and it may or may not be in subsegment R . If these events were independent we would expect R_{truth} , the number of ground truth pixels in R to be centered in the range $LB \leq R_{\text{truth}} \leq UB$. Rather, we expect a correlation among these events. If S contains both truth and non-truth pixels then the truth pixels will have a bias toward one subsegment and the non-truth pixels to the other.

To verify this hypothesis we observe $P(R_{\text{truth}}|S_{\text{truth}}, S_{\text{size}}, R_{\text{size}})$, the conditional distribution of ground truth intersection of a subsegment on a parent segment. This data was obtained from segmentations computed by the algorithm given in Section 3. These histograms plot the size ratio of intersections between ground truth segments and images segments R scaled linearly over the feasible range $\frac{R_{\text{truth}} - LB}{UB - LB}$. We only consider cases where $R_{\text{size}} > 40$ and $R_{\text{size}} = S_{\text{size}} \pm 10\%$.

The histograms in Figures 5.1 and 5.2 indicate that when S contains some ground truth pixels, they appear disproportionately in either the left or right subsegment.

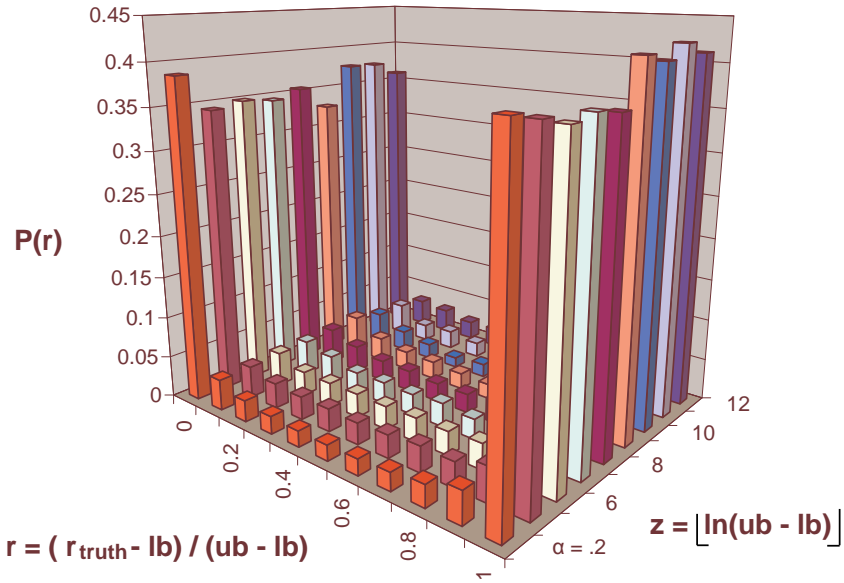


Figure 5.1: The intersection of subsegments and ground truth. Regardless of segment size (z axis), the ground truth pixels in segment S tend to fall in either the right or left subsegments. The distribution is closely approximated by the Beta distribution (foremost, in red).

We suggest to fit a Beta distribution over the relative sizes of L and R :

$$B(x, \alpha, \beta) = \frac{1}{B(\alpha, \beta)} x^{\alpha-1} (1-x)^{\beta-1} \quad (5.2.3)$$

which is the distribution of the $(\alpha+1)$ th highest value of $\alpha+\beta+1$ samples of the uniform distribution over $[0, 1]$ and $B(\alpha, \beta)$ is the normalizing beta function.

Given that $L_{\text{truth}} + R_{\text{truth}} = S_{\text{truth}}$, we model the probability density or relative segment size size using the Beta distribution (Figure 3.1)

$$q_{\text{truth}}(S, R, L, \omega_{\text{truth}}) = c \left(\frac{R_{\text{truth}} - LB}{UB - LB} \right)^{\omega_{\text{truth}} - 1} \left(\frac{UB - R_{\text{truth}}}{UB - LB} \right)^{\omega_{\text{truth}} - 1}, \quad (5.2.4)$$

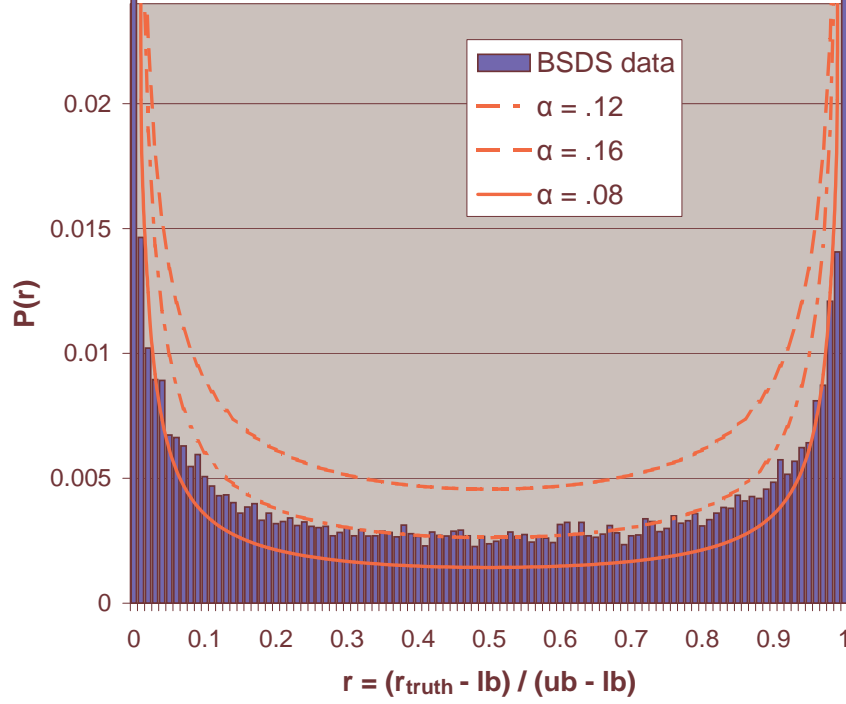


Figure 5.2: The intersection of subsegments and ground truth. A histogram of 100 bins over the possible values for R_{truth} is closely approximated by the Beta distribution.

where c is a constant. For

$$\omega_{\text{truth}} = \alpha = \beta,$$

the expected value will be at

$$\mathbb{E} \left(\frac{L_{\text{truth}}}{S_{\text{truth}}} \right) = \frac{\alpha}{\alpha + \beta} = \frac{1}{2}.$$

Also, for

$$\omega_{\text{truth}} = \alpha = \beta < 1.0,$$

the distribution becomes U-shaped, with maxima in the limits of LB and UB . Therefore, the ground truth pixels of segment S will tend to appear in the left subsegment or right subsegment, but not both.

Figure 5.1 demonstrates invariance to segment size. Figure 5.2 shows the probability mass function in greater detail.

5.3 Computing the Segment Description Length

In this section, we give an efficient algorithm for evaluating the proposed segmentation model using cross entropy.

Let segmentation S be a set of three-tuples of segments $\{S, L, R\}$, such that R and L are the left and right children of S in a hierarchical model. Let $Y = \{Y_i\}, 1 \leq i \leq M$ are the hidden variables, and $X = \{X_i\}, 1 \leq i \leq N$ are the pixel variables. Segment variables take values in $\mathbb{T} \cup \mathbb{S}$. Given S , computation of Y is straightforward: we simply take $S_{\text{truth}} = L_{\text{truth}} + R_{\text{truth}}$. Under this assumption, we may take

$$\begin{aligned} q(Y_0) \prod_{\{S,L,R\} \in \mathbf{S}} q_{\text{truth}}(S, L, R, \Omega) \\ = q(Y_0, Y_1, \dots, Y_n, X_0, X_1, \dots, X_n) \\ = q(X_1, X_2, X_3, \dots, X_N) \end{aligned} \quad (5.3.1)$$

which is the ground truth distribution estimate. $q(Y_0)$ is the prior on the root segment which we take to be a constant function.

Suppose we were to describe the state of X in a language of b symbols. A description is manifested as a string of symbols that uniquely identifies the state of X_i for all $1 \leq i \leq N$. In a hypothetical descriptive language of b symbols, the optimal length of a description for state X is

$$\log \left(\frac{1}{P(X)} \right). \quad (5.3.2)$$

Suppose we precompute an optimal descriptive language based on the the estimate, $q(X)$. The the description length is then given by

$$\log \left(\frac{1}{q(X)} \right). \quad (5.3.3)$$

We may perform the factorization

$$\begin{aligned} \log_b \frac{1}{q(X)} &= \log_b \left(\frac{1}{q(X_1, X_2, X_3, \dots, X_N)} \right) \\ &= -\log(q(X_1, X_2, X_3, \dots, X_N)) \\ &= -\log(q(Y_0) \prod_{\{S,L,R\} \in \mathbf{S}} q_{\text{truth}}(S, L, R, \Omega)) \\ &= c - \sum_{\{S,L,R\} \in \mathbf{S}} \log_b(q_{\text{truth}}(S, L, R, \Omega)) \end{aligned} \quad (5.3.4)$$

The factorization reveals that the description can be broken into parts, such that each part describes the number of ground truth pixels in a subsegment, given the number in the parent. The length of the total description is the sum of the descriptions of the parts.

In the segment description algorithm, we implement a recursive strategy to compute the length of the description for a ground truth segment. This strategy is exemplary of the top-down generative process of image formation, and a proper descriptive language might implement the strategy. Given an image and a segmentation, we first specify how many of the ground truth pixels are in each of the top-level segments. Then for each of these segments, describe how many pixels are each of its subsegments. Repeat until the leaf level is reached. The transmission of fractional bits which is not possible in digital communication yet the top-down approach is correct because children are independent given the state of their lowest common ancestor. There will be less uncertainty in the state of lower-level segments since, if the segmentation was computed correctly, the pixels of those segments are increasingly ‘alike’.

Consider the distribution

$$p(X_1, X_2, X_3, \dots, X_N) \quad (5.3.5)$$

of ‘distinguished things’ in the Berkeley Segmentation Dataset (BSDS) [44] ground truth segmentations. We might call this the sampled ground truth segment distribution. Although the BSDS is given in the form of partition segmentations, we consider the data set as a ‘bag of segments’ $V = \{\mathbf{v}^j\}, 1 \leq j \leq K$. K is the number of ground truth segments for all subjects, given a specific image. Each ground truth segment is given by $\mathbf{v}^j := [\mathbf{v}_i^j], 1 \leq i \leq N$ where $\mathbf{v}_i^j = 1$ indicates the i th pixel is ‘inside’ the j th segment and $\mathbf{v}_i^j = 0$ indicates otherwise. Therefore, instead of computing the summation over the entire 2^N domain of p , we sum over the K sampled ground truth segments.

Algorithm 3 (Segment Description Algorithm)

Let $X = \{X_i\}, 1 \leq i \leq N$ be segment variables taking values in $\mathbb{T} \cup \mathbb{S}$.

Let \mathcal{L} be the recursive function

$$\mathcal{L}(S, S) := \begin{cases} \log_2(q_{\text{truth}}(S, L, R, \omega_{\text{truth}})) + \mathcal{L}(L, S) + \mathcal{L}(R, S) & \text{if } \{S, L, R\} \in \mathbb{S} \\ 0 & \text{otherwise} \end{cases} \quad (5.3.6)$$

where S, L and R are image segments and segmentation S is a set of three-tuples of segments $\{S, L, R\}$, such that R and L are the left and right children of S in a hierarchical model.

Let V be a set containing K ground truth segments.

1. *Initialize X directly from an image. Segment X via the region merging algorithm to obtain segmentation S . Let Y_{root} be the root segments in S .*
2. *For each \mathbf{v} in V ,*
 - (a) *Initialize $X_{i_{truth}} = \mathbf{v}_i, 1 \leq i \leq N$.*
 - (b) *Compute $S_{truth} = L_{truth} + R_{truth}, \forall \{S, L, R\} \in S$*
 - (c) *Compute the description length*

$$DL(S, \mathbf{v}) = \mathcal{L}(Y_{root}, S), \quad (5.3.7)$$

We divide the average description length by the shortest possible description length we may obtain without observing pixel correlations (Equation 5.1.8). Therefore, we report the amount of compression we obtain over the baseline method. This normalizes the measure with respect to N , the image size.

Hopefully, a hierarchical decomposition may still achieve a good estimate of the true joint probability distribution. What are the sources of error? At each subdivision we summarize the probability of ground truth state for pixels in S by S_{truth} , the number of pixels which are ‘truth’. Unless nature is a perfect hierarchy this will induce some error. Of course, even if nature is a perfect hierarchy there are still two sources of error: our ground truth probability estimate (Section 5.2) or our approximation of the internal structure (Algorithm 1).

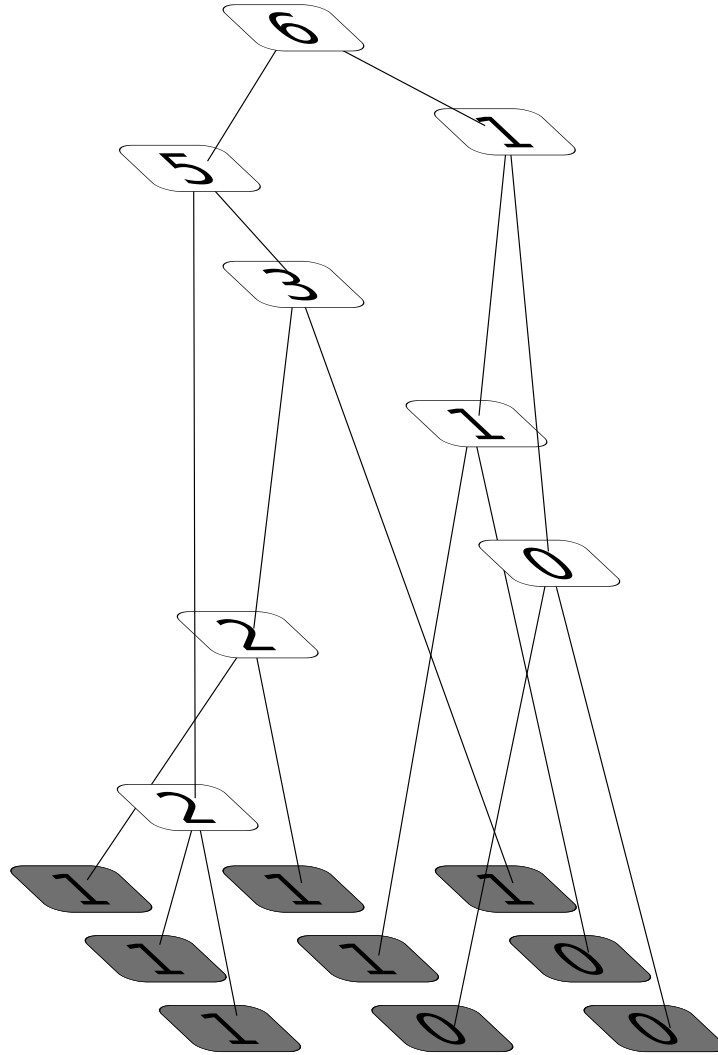


Figure 5.3: The segment description algorithm. Labels indicate the number of ‘ground truth’ pixels within the segment for a single ground truth segment selected from the ground truth probability distribution. Observations show (Figure 5.1) that ground truth pixels will appear in one ‘child’ or the other of a segment. We combine the structure computed by the region merging algorithm (Figure 3.3) with the beta distribution (Figure 5.2) to estimate the joint probability distribution of pixel ground truth state.

Chapter 6

Experimentation

6.1 Evolving the Descriptive Language for Reduced Description Length

The description length evaluation method allows to evaluate and compare the performance of segmentation algorithms. The ability to quantify the results of a scientific proposal is crucial, yet “the usefulness of such measures is not confined to comparing different approaches, but provides an important step to building self-optimizing vision systems that automatically adjust algorithm parameters at each level of the system to improve performance [52]”.

In the case of image segmentation we often encounter segmentation criteria which are parametrized in some way. For instance the criterion may combine many factors that suggest the presence of objects. A variational criterion must weigh the relative influence of those factors parametrically. The parameters of our algorithm are the vector

$$\Omega = [\omega_{\text{size}} \omega_{\text{luminance}} \omega_{\text{colour}} \omega_{\text{shape}} \omega_{\text{truth}}], \quad (6.1.1)$$

which controls all adjustment of the hierarchical model. While the segmentation criterion is formulated from our assumptions about the real world, ‘tuning’ the parameters of the criterion will take more care.

Reviewing the methods of Chapter 2, we see many authors agree that segmentation can be improved by using multipart criteria. However they disagree as to what those factors should be and how they should be defined. In most cases, presumably, the weights and thresholds by which these authors combine criteria are chosen according to their own intuition.

Tuning the parameters of the algorithm poses a nonlinear multivariate optimization problem. This is a difficult problem because while we can evaluate for a single point in search space (the space defined by the parameters), we cannot evaluate the derivatives of the space. We cannot solve for maxima or even perform gradient-ascent type searches. It can be difficult to find the optimal value in such a case especially when the function is noisy.

We employ the **evolution strategies** (ES) optimization method [11]. ES is an optimization method for unknown functions with real-valued, high-dimensional search spaces. The evolutionary strategies method of optimization (Algorithm 4) uses the biologically motivated principles of variation and selection to evolve a population of search points into continually better regions of the search space.

ES is appropriate for use in a search space that is high-dimensional and real-valued, and when the objective value is noisy and no gradient information is possible. We consider the compression ratio to be a noisy objective function since it is complex and is computed on a relatively small training set.

Algorithm 4 (Evolution Strategies)

1. *Initialize population*

$$\mathbf{X}_i = \Omega_i, 1 \leq i \leq \mu.$$

2. *Selection: Test the fitness of each individual. Segment each image according to the parameter set \mathbf{X}_i . Average over all images the cross entropy between the sampled ground truth probability distribution p and the estimated ground truth probability distribution q .*

$$\mathbf{Y}_i = H(p, q).$$

Select ρ of the best results according to the selection operator.

3. *Recombination: Combine parents \mathbf{X}_i and \mathbf{X}_j into new search point \mathbf{X}_C , according to the recombination operator.*

4. *Mutation: Add a random, normally distributed vector.*

$$\mathbf{X}_C = \mathbf{X}_C + \sigma N(0, I)$$

5. *Repeat from recombination to produce λ children.*

6. *Repeat from selection until some stopping condition*

We use a population size of 10 solutions, each solution representing a vector of parameters Ω_i . For each solution we compute the segmentation of each image, and for each image, compute the cost to transmit all ground truth segments in terms of the segmentation of the image. We sum the number of bits in the description length for the hierarchical approach versus the method using a constant number of bits (≈ 0.286 , see Section 5.3) for each pixel. The ratio between the two is averaged over all images to yield the fitness of the solution.

A purely biologically based implementation might select two highly fit individuals and use intermediate recombination - take their mean - to produce an offspring. We employ the **optimal weighted recombination** (OWR) recombination operator [4], where all individuals contribute to an offspring with genetic contribution weighted by their relative fitness.

An attractive property of the ES system is self-adjusting mutation strength [28]. Unlike simulated annealing which generally has a predetermined plan (annealing schedule) for gradually reducing the mutation strength, ES attempts to adjust this parameter dynamically. The general approach is to accumulate the progress made over the last few generations and to use this data to infer the trajectory of future generations. In cumulative step-length adaptation, if the cumulative path is relatively long then the previous steps are correlated. This indicates that the same amount of progress could have been made in fewer steps if the mutation strength were larger. In this case, the mutation strength σ is correspondingly increased. On the other hand if the progress vector (recent search path) is relatively short over the last few steps it suggest that those steps have been negatively correlated. In other words the previous steps have been ‘doubling back’ on themselves, indicating that the mutation strength is too high. Furthermore, the progress vector can also be used to adjust the covariance matrix of the mutation operator increasing the probability of mutating into successful directions.

By exploring the search space with the evolutionary search algorithm, we hope to observe the relative importance of different terms in the criterion and ultimately achieve the best possible ground truth compression.

Because all parameters must be positive non-zero, and because more precision is needed near zero, optimization is done in the (natural) log domain.

6.2 Results

Possibly due to the relatively small dimensionality of the search space, our attempts to incorporate step size adaptation were unsuccessful. Instead we fixed the mutation strength σ to a small value, resulting in stable optimization runs at the expense of progress rate.

Due to time constraints, optimizations were performed using a subset of twenty of the 200 images in the training set.

Several optimization runs were performed from different starting locations. Initial tests appeared to converge toward the vicinity of

$$\begin{aligned}\Omega &= [\omega_{\text{size}} \quad \omega_{\text{luminance}} \quad \omega_{\text{colour}} \quad \omega_{\text{shape}} \quad \omega_{\text{truth}}] \\ &= [\exp(1.5) \quad \exp(-1.0) \quad \exp(1.5) \quad \exp(-1.0) \quad \exp(-1.5)],\end{aligned}$$

the estimated minimum. Searches starting nearby quickly approached the estimated minimum and then performed a random walk of the region. A search beginning far from the estimated minimum progressed more slowly through the search space. These behaviours may both be due to an incorrectly tuned mutation strength. It also appears that near-best compression results can be obtained in the vicinity of

$$\begin{aligned}\Omega &= [\omega_{\text{size}} \quad \omega_{\text{luminance}} \quad \omega_{\text{colour}} \quad \omega_{\text{shape}} \quad \omega_{\text{truth}}] \\ &= [\exp(-2.00) \quad \exp(-0.35) \quad \exp(3.15) \quad \exp(-0.68) \quad \exp(-1.06)],\end{aligned}$$

suggesting the ‘fitness landscape’ is more flat than previously thought. Segmentation results using these two vectors are compared visually in Figure 6.1. Note that among these two vectors the relative importance of ω_{size} and ω_{shape} are reversed. During optimization, it was observed that one of these terms tends to increase as the other decreases. This may be explained by the fact that perimeter and size are, after all, dependent. Overall best compression results were just under 2%.

Visual segmentation results are given in Figures 6.4 to 6.7 alongside the ground truth segmentations. While we do not observe automatically generated segments (left column) that match exactly to hand-generated segments (right column), it seems we can obtain the shape of the hand-generated segments by taking the union of automatically generated segments at some (high) level in the hierarchy.

Figure 6.4 (BSDS image #223061) shows the Louvre Pyramid, a glass structure which is difficult to segment due to the transparency of glass objects. The algorithm has difficulty delineating the thin metal structures diagonal to the pixel grid. Yet at a high level the overall structure of the building is apparent.

Figure 6.5 (BSDS image #105025) shows a lion in a natural habitat. At high levels the scene is partitioned into distinct coloured regions but at intermediate levels the algorithm seems to have difficulty finding the boundaries between textured regions. Note that two subjects delineated the lion's eye and nose despite the very small size of these objects.

Figure 6.6 (BSDS image #376020) shows a monk sitting by a tree. The colouration of the image subject allows for effective segmentation. Smaller objects such as the white strip in the background and the yellow material in the foreground are also effectively segmented. Note, though, that near the top of the hierarchy the subject's head is joined to the tree rather than to the rest of his body.

Figure 6.7 (BSDS image #188005) shows a person on a boat in front of an iceberg. The segmentation produced by our algorithm seems reasonable and at level 16 most salient regions are grouped.

Processing time for a single 480x320 image averaged under six seconds on a Pentium Core 2 Quad processor at 2.4 GHz. Note that the majority of processing time is spent on very small segments. A preliminary segmentation algorithm, such as the watershed algorithm or an algorithm that merges on locally maximal criteria values [23] could improve running time substantially.

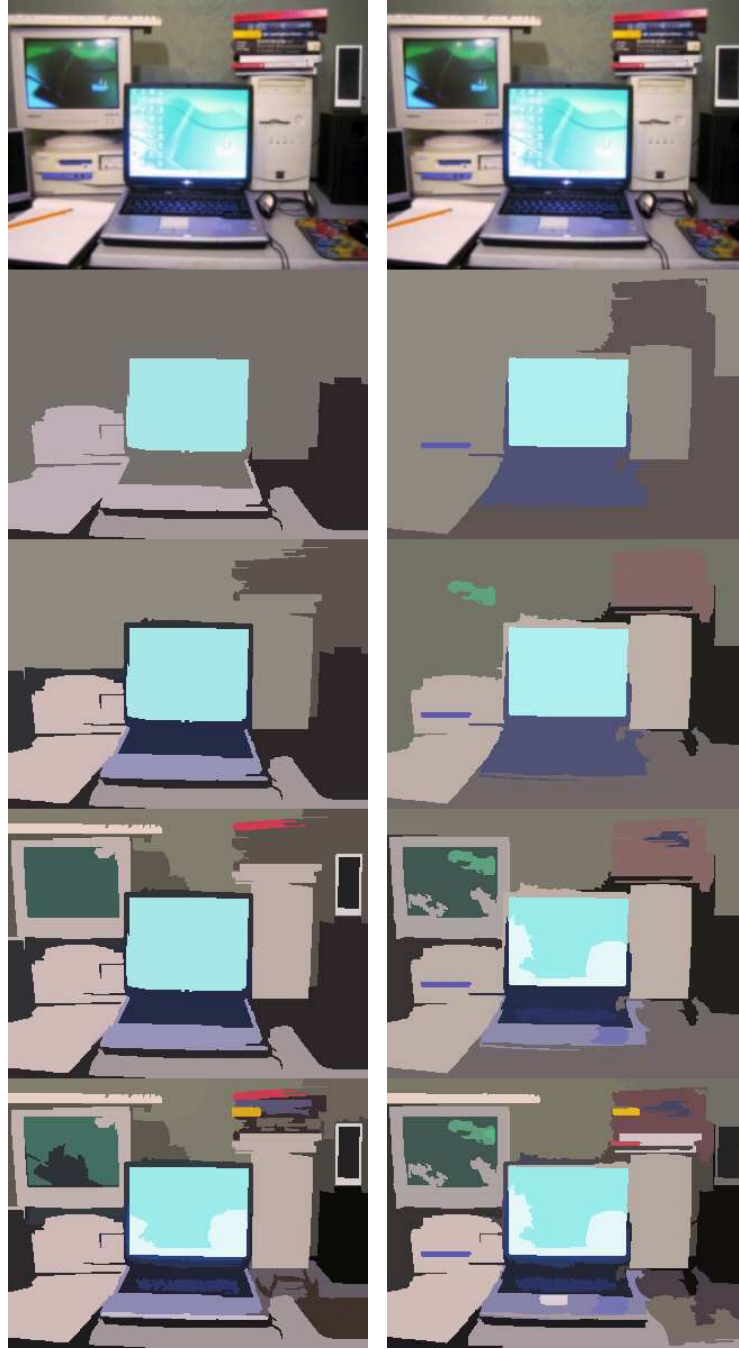


Figure 6.1: Left column: Segmentation generated by parameters at the estimated minimum, $\exp(\Omega) = [1.5, -1.0, 1.5, -1.0, -1.5]$. Right column: segmentation generated at alternate parameter settings $\exp(\Omega) = [-2.00, -0.35, 3.15, -0.68, -1.06]$ (see Figure 6.2). The topmost level contains the same desk image. Subsequent levels display the segment hierarchies at 4, 8, 16, and 32 segments.

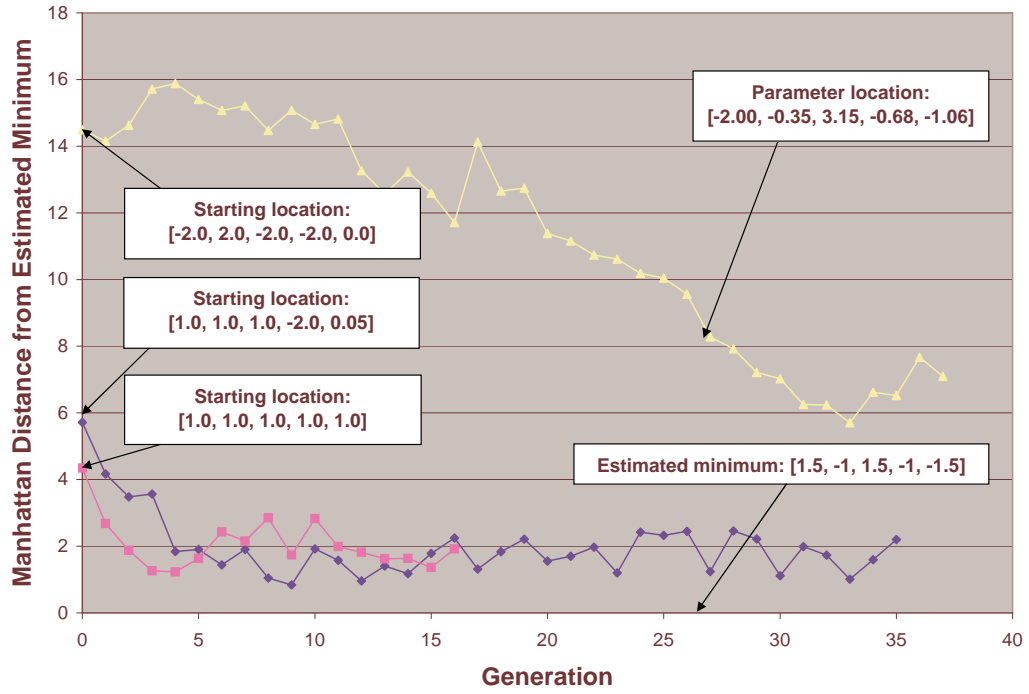


Figure 6.2: Position in search space approaches the estimated minimum as optimization progresses. Three optimization runs are shown from different starting locations.

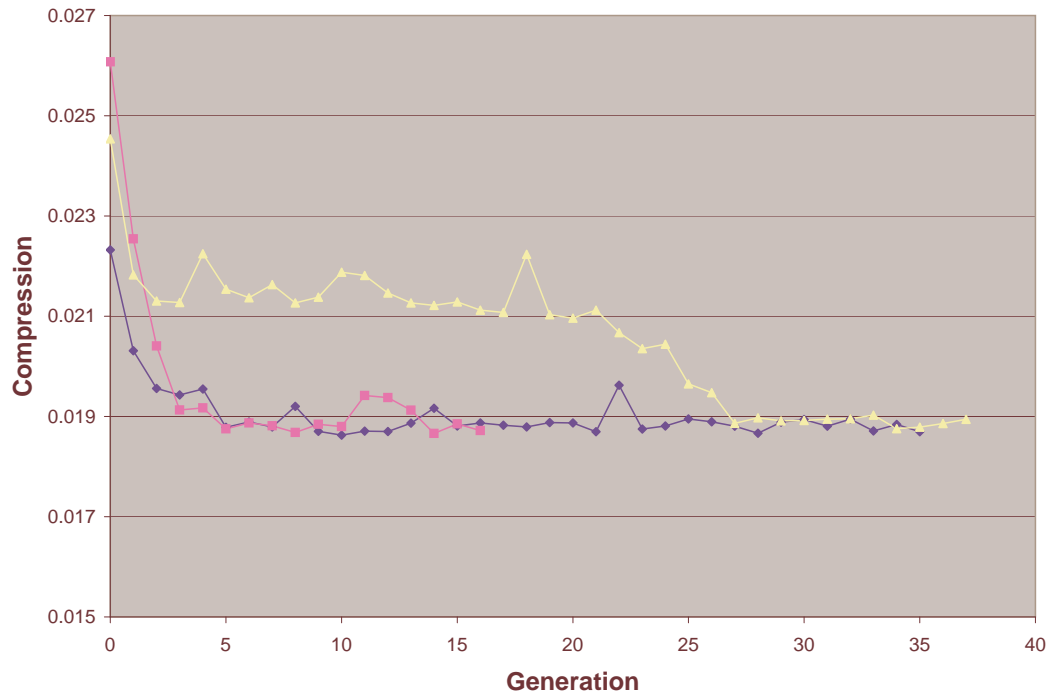


Figure 6.3: Description length of ground truth decreases as optimization progresses. Three optimization runs are shown from different starting locations.

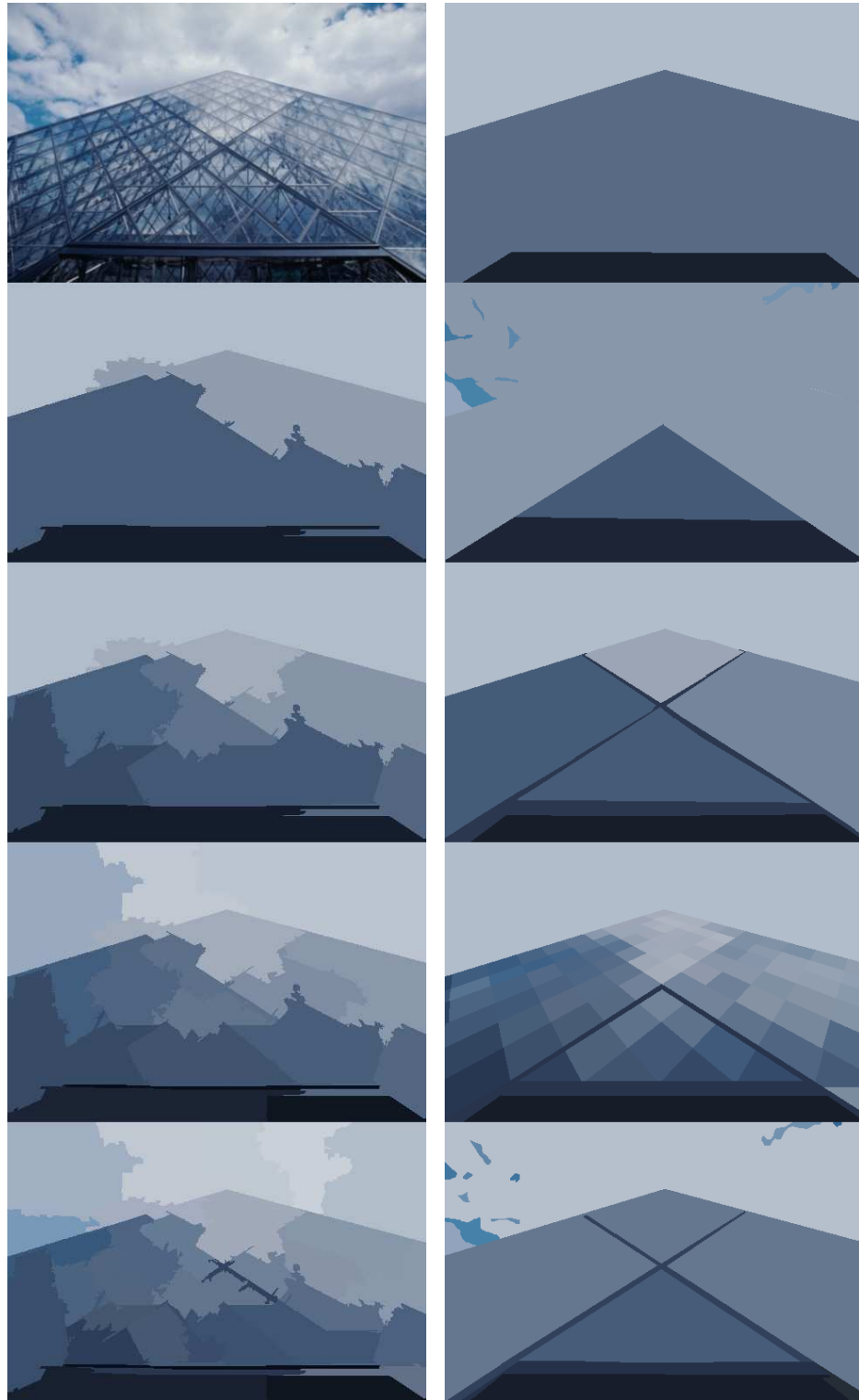


Figure 6.4: Left column: a digital image and its automatic segmentation at levels 4, 8, 16, 32. Right column: ground truth segmentations by different human subjects.



Figure 6.5: Left column: a digital image and its automatic segmentation at levels 4, 8, 16, 32. Right column: ground truth segmentations by different human subjects.



Figure 6.6: Left column: a digital image and its automatic segmentation at levels 4, 8, 16, 32. Right column: ground truth segmentations by different human subjects.

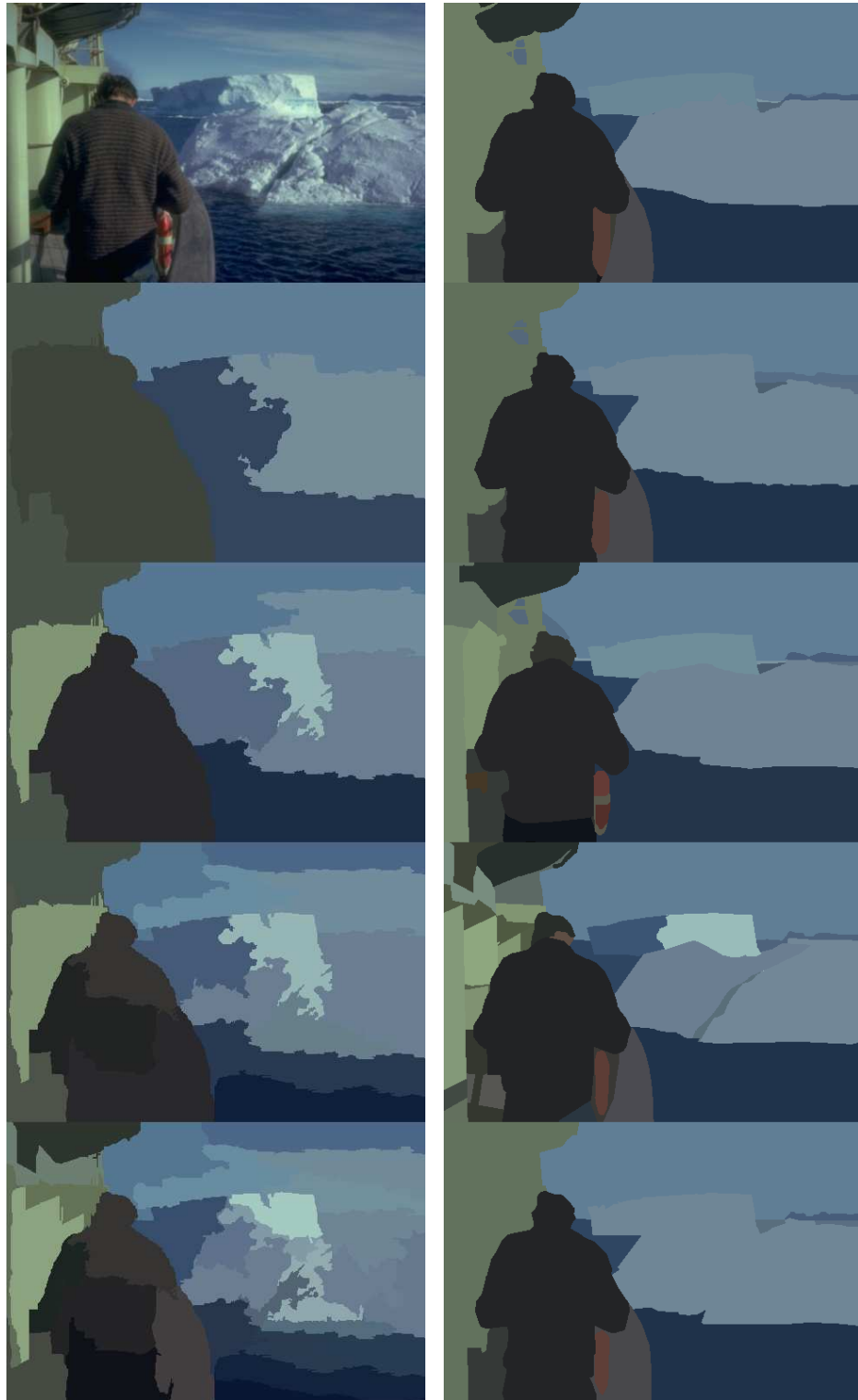


Figure 6.7: Left column: a digital image and its automatic segmentation at levels 4, 8, 16, 32. Right column: ground truth segmentations by different human subjects.

Chapter 7

Applications

Two applications were developed using the optimized segmentation algorithm. HSVGen, a bitmap-to-SVG image converter, is an example of a ‘visualization’ application and SAFE, a semi-automatic foreground extractor, is an example of an ‘analysis’ application. Both are made available for download on the author’s web site [39].

7.1 HSVGen: A Hierarchical Bitmap-to-SVG Image Converter

We developed software based on the proposed segmentation algorithm to convert between bitmap and vector image types [40]. A vector ‘graphic’ is composed of points, lines and polygons, and in its digital form is more precise than a bitmap image under resizing and rotation. While bitmap images are easily acquired by digital photography, vector types are often the principal format used for graphic design. Our intention was a bitmap to vector converter that would respect the organization of objects in the scene, not only their appearance.

The conversion draws a polygon for each image segment at some specified level. Polygon boundaries are then refined to subpixel precision using an active contour formulation with region competition [57, 81]. Segment boundaries are also simplified for smaller file sizes using the quadric mesh decimation method of Garland and Heckbert [25], which has interesting parallels with the region merging algorithm.

The application uses .NET components, accepts JPEG, GIF and other bitmap formats, and saves the conversion result in **scalable vector graphics** (SVG) format [27] (See Figure 7.1).

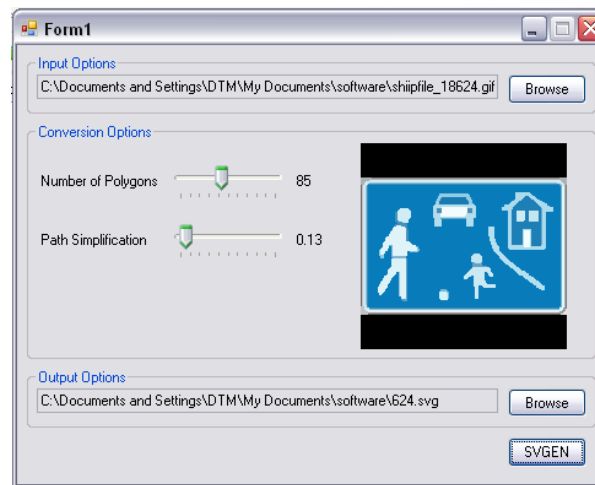


Figure 7.1: HSVGEN: a raster-to-vector conversion program.

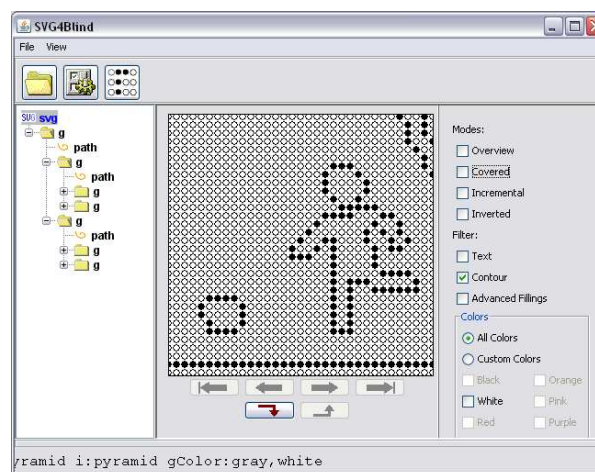


Figure 7.2: SVG4Blind: a tactile image explorer.

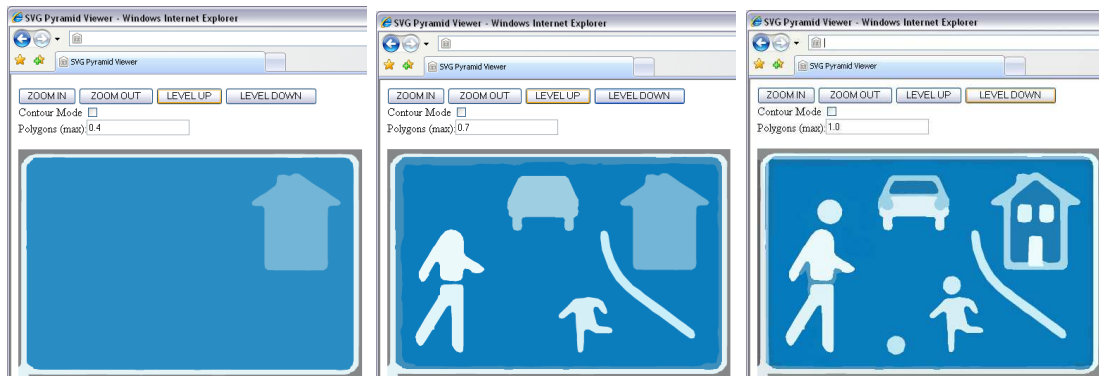


Figure 7.3: A scalable vector graphic (SVG) with interactive level-of-detail.

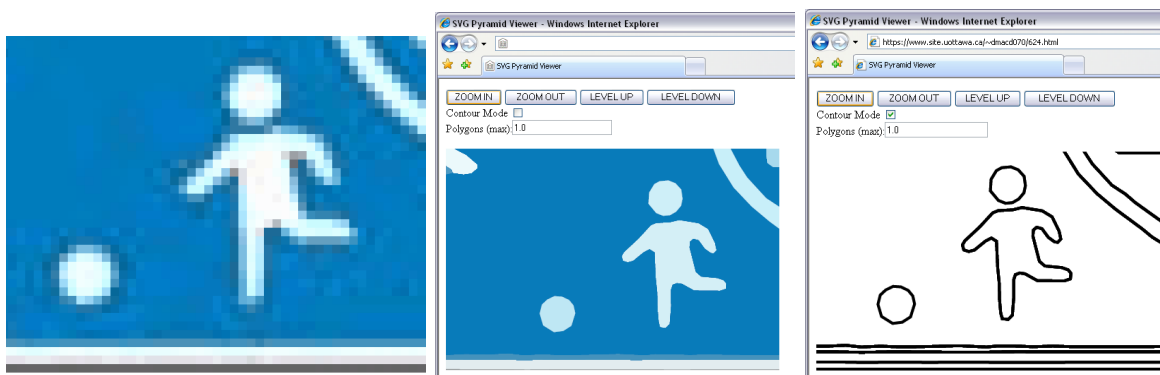


Figure 7.4: Image detail of 7.4(a) a poor quality bitmap 7.4(b) image segments computed by the proposed algorithm 7.4(c) segment contours

Because it is based on a hierarchical algorithm, we are able to produce an SVG with multiple levels of detail. SVG is an extensible markup language (XML) format which is ideally suited for the hierarchical structure. We add an attribute ‘detail’ to the principal container element ‘group’, indicating the level of polygon in the hierarchy. Then, the rendering of detail can be controlled programmatically. We use Javascript embedded in HTML to filter out any polygons which are below the detail threshold. Figure 7.3 shows the results of a low-quality bitmap that is converted into a hierarchical SVG. Note that objects have largely been separated into distinct polygons. Furthermore, note that the windows of the house and the headlights of the car are proper subsegments of those larger objects. A detailed view reveals the effectiveness in extracting object boundaries (Figure 7.4(c)), despite compression artifacts in the original (Figure 7.4(a)).

This application demonstrates interesting possibilities for image visualization. Our software was incorporated into the SVG4Blind system [58], a software for rendering images on a tactile display. The extraction of contours and multiple levels of detail were found to aid image exploration on the low-resolution, monochrome device (Figure 7.2).

7.2 SAFE: A Semi-Automatic Foreground Extractor

Often, a digital photograph contains a ‘subject’, which is a region of the image that is the focus of interest. The background of an image is any pixels that are not part of the subject, but that have been captured in the image by circumstance. To perform a visualization or analysis task on the subject alone it is necessary to properly label those pixels. Accordingly, we developed a tool which assists a user to subtract background pixels of an image, leaving only the foreground or ‘subject’ of an image.

A commercially-available example of such a tool is the Adobe(R) Photoshop(R) Magnetic Lasso tool. This tool is ‘especially useful for quickly selecting objects with complex edges set against high-contrast backgrounds’ [1]. Using the assumption that users generally extraction regions which correspond to distinct objects, and with the observation that there is often a steep transition in pixel appearance at the boundary of two objects, the Magnetic Lasso tool assists the user by refining a rough estimate of the boundary of the desired region.

Our approach is region-based rather than boundary based. The SAFE application allows a user to subdivide the pixels of an image into two classes, foreground and background (or any other bipartition), based on the assumption that if two pixels are close in colour and proximity, they they will likely be both foreground or both background.

Essentially, the SAFE application allows the user to to ‘describe’, or uniquely identify, which pixels of the image correspond to the subject of that image. Therefore the task conveniently illustrates the framework of optimization proposed in this thesis.

The application begins by segmenting the image to be processed using the region-merging algorithm. Selection is then performed in a top-down manner. Initially, the image is divided into the two topmost segments. A white line designates the boundary between segments. The user can give a command to further subdivide the segments, down to the level at which each segment contains only one pixel. Subdivision is simply the opposite of merging, and we consider the traversal ‘top-down’ on the hierarchy.

At any time, the user may use a pointing device to select a segment. All pixels within the selected segment are labeled ‘foreground’. Depending on the image, a user will often have to subdivide the image more than once before segments appear which are wholly ‘foreground’. If those segments appear at high levels in the segment hierarchy then less user interaction will be required. Therefore, if the segmentation algorithm is able to merge together small segments which correspond only to the subject of the image, then the user will have to perform fewer selections.

The process of selecting the subject of the image is strongly related to the segment description algorithm. Instead of communicating the number of ground truth pixels in a segment, we indicate that the number of ‘background’ pixels in the segment is zero. The implication for segmentation is the same: if a segment contains some foreground and some background pixels, then we hope that when the segment is subdivided the background pixels will appear only in one subsegment or the other. Therefore as the discrepancy between the ground truth and the segmentation improves, fewer selections will have to be made by the user.

The application accepts standard bitmap image types and saves results in the portable network graphics (PNG) format. Background pixels are coloured ‘transparent’, so that the subject can be composited onto a different background (Figure 7.2).

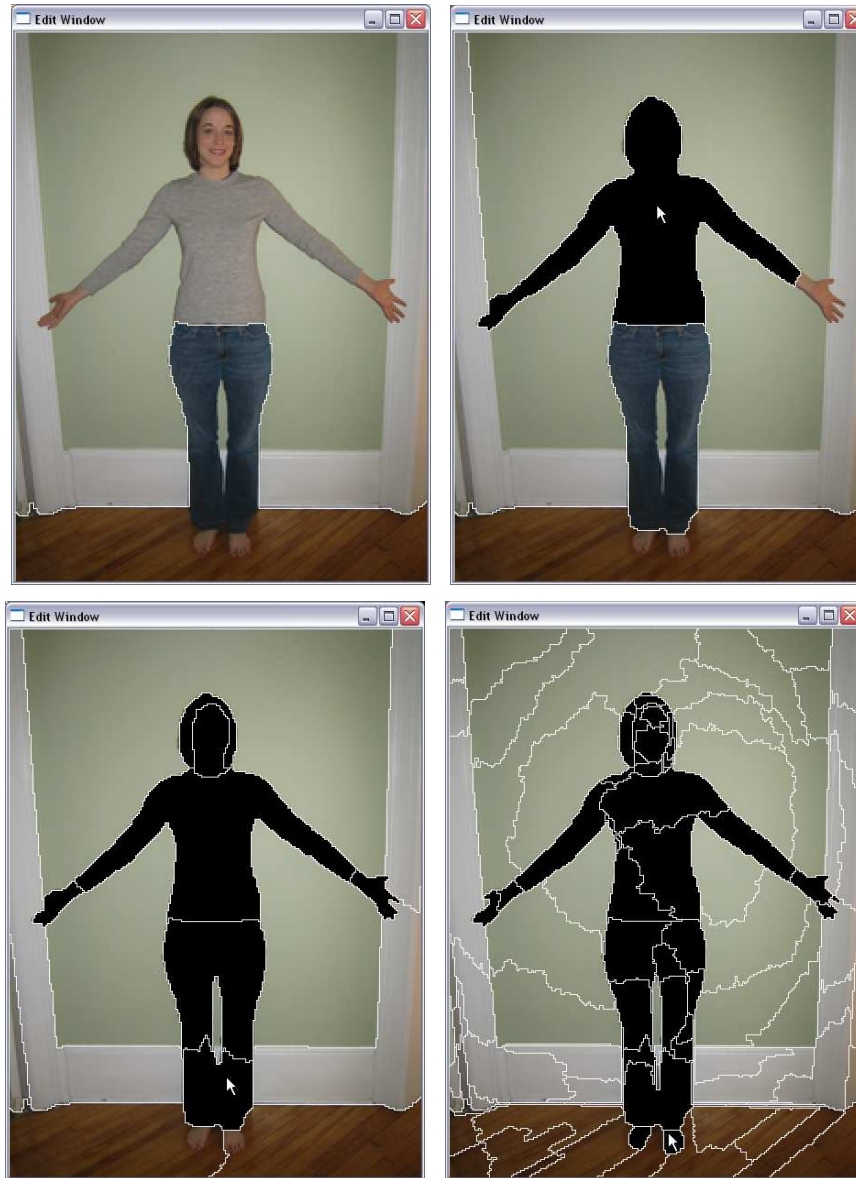


Figure 7.5: Background subtraction with SAFE. Figure 7.5(a): The first subdivision is not able to segment the image into subject and background. Figure 7.5(b): After another subdivision, a segment appears which only contains subject pixels. Figure 7.5(d): After several more subdivisions, the final subject pixels may be selected.



Figure 7.6: An image composite of several subjects extracted with SAFE. Image Copyright 2007 WorldHoldingHands.

Chapter 8

Conclusions and Future Work

In this thesis, we present a method for hierarchical segmentation of an image as a preprocessing step for image visualization or analysis. The hierarchical model allows to approximate multi-part objects and offers different levels of abstraction of the image. We show that the data structure can predict a separate distribution: human-delineated, ground truth segments of ‘distinguished things’. The hierarchical model is able to uniquely identify such segments in approximately 2% of the description length used by a baseline assumption.

Used as an evaluation device this performance characterization gives a substantive measure by which to trade-off accuracy and performance. In comparison most of the measures reviewed in Chapter 2 have no context beyond the one test they perform. For example, the γ measure used in our previous work gave higher values for the more expensive earth mover’s distance [41] criterion, but it was difficult to understand *how much* better the results were or whether it warranted the additional computational expense. Meilă [45] also notes that the information-based approach has better comparability over different experimental conditions.

This work is important because while hierarchical segmentation algorithms are available in ever increasing variety, “few benchmark databases and standard evaluation procedures exist, and selecting the best algorithm for a particular application is usually a matter of trial and error [69].” It is hoped that this thesis will inspire more work into quantitative evaluation of hierarchical image segmentation criteria.

The criterion used in this work is meant to be exemplary rather than definitive. LeClerc states, “The advantage of [the minimum description length] formulation is that it can be extended to deal with subsequent steps of the image understanding problem

(or to deal with other attributes, such as texture) in a natural way by augmenting the descriptive language [36].” We could conceivably add terms for texture, boundary gradient [68], optical flow, higher-level object knowledge [82], and so on. In the near future we hope to experiment with more complex colour criteria, incorporating the comparison of colour distributions by parametric [41] or non-parametric [43] means, rather than simply comparing mean colour values.

More theoretical study into the distribution of visual and other features of a segment hierarchy is required. Most pressing, the size and truth-intersection terms are not properly understood. The observation that the size of ground truth segments follows a power law may be a good starting point [44]. Until then, we are satisfied with the Beta distribution. Tractable numerical approximations of the function are available [54].

To our knowledge, this thesis is the first effort to model the joint inclusion distribution of BSDS ground truth segments hierarchically. We have attempted to show that the approach is straight-forward, reproducible and useful. We do not experimentally compare the hierarchical model to the piecewise constant or piecewise smooth models. We argue that higher compression is possible with a hierarchical model, because the BSDS seems to contain images of hierarchically organized objects. Of course, to disprove this argument, it would be sufficient to show that better compression can be obtained via another model.

The hierarchical image model presents interesting possibilities for image display and analysis. We developed software based on the proposed segmentation algorithm to convert between bitmap and vector image types (See Figure 7.1). Our intention was a raster to vector converter that would respect the organization of objects in the scene, not just appearance. Other applications in visualization include object-based video coding and object-based interactivity of SVG graphics. One reason for the difficulty in building effective computer vision systems such as object recognition is the quantity of data that must be processed. We have shown that by simply grouping similar pixels we may improve the efficiency of this particular computer vision task substantially. It is reasonable to suspect that others may benefit as well.

A number of experimental setups may be envisioned under the proposed evaluation framework. In this thesis we experiment with different weighting factors for a multi-term segmentation criteria. But, we could also experiment with different criteria, such as those involving texture. It would be interesting to compare the performance of the hierarchical model versus the piece-wise smooth and piece-wise constant models. Algorithms other than region-merging could be tested. The approach could also be applied to other data sets, such as the viewpoint stability set [41].

Apart from maintaining good visual fidelity to the original image we believe our method has great potential for object-based visualization and analysis. A concise and simple description of objects is important for the efficiency and robustness of computer vision applications [59]. Moreover from an information theoretical perspective a concise description demonstrates an accurate understanding of the underlying distribution, in this case, of objects in the scene.

Bibliography

- [1] Adobe Help Resource Center. Select with the magnetic lasso tool. http://livedocs.adobe.com/en_US/Photoshop/10.0/, 2008.
- [2] M. Aghagolzadeh, H. Soltanian-Zadeh, B Araabi, and A Aghagolzadeh. A hierarchical clustering based on mutual information maximization. In *Proceedings of the IEEE International Conference on Image Processing*, volume 1, pages 277–280, 2007.
- [3] C. Ané and M. J. Sanderson. Missing the forest for the trees: Phylogenetic compression and its implications for inferring complex evolutionary histories. *Systematic Biology*, 54:146–157, 2005.
- [4] D. V. Arnold. Optimal weighted recombination. In *International Workshop on Foundations in Genetic Algorithms*, pages 227–237, 2005.
- [5] F. B. Baker. Stability of two hierarchical grouping techniques case I: Sensitivity to data errors. *Journal of the American Statistical Association*, 69(346):440–445, 1974.
- [6] F. B. Baker and L. J. Hubert. Measuring the power of hierarchical cluster analysis. *Journal of the American Statistical Association*, 70(349):31–38, 1975.
- [7] J.-P. Barthélemy, B. Leclerc, and B. Monjardet. On the use of ordered sets in problems of comparison and consensus of classifications. *Journal of Classification*, 3:187–224, 1986.
- [8] J.-M. Beaulieu and M. Goldberg. Hierarchy in picture segmentation: A stepwise optimization approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(2):150–163, 1989.

- [9] E. Ben-Hur, A. Elisseeff, and I. Guyon. A stability based method for discovering structure in clustered data. In *Proceedings of the 7th Pacific Symposium on Biocomputing (PSB '02)*, volume 7, pages 6–17, January 2002.
- [10] J. R. Beveridge, J. Griffith, R. R. Kohler, A. R. Hanson, and E. M. Riseman. Segmenting images using localized histograms and region merging. *International Journal of Computer Vision*, 2:311–347, 1989.
- [11] H. G. Beyer and H. P. Schwefel. Evolution strategies - a comprehensive introduction. *Natural Computing*, 1(1):3–52, 2002.
- [12] D. M. Blei and M. I. Jordan. Variational inference for dirichlet process mixtures. *Bayesian Analysis*, 1(1):121–144, 2006.
- [13] S. Borra and S. Sarkar. A framework for performance characterization of intermediate-level grouping modules. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:1306–1323, 1997.
- [14] T. Brox, D. Farin, and P. H. N. deWith. Multi-stage region merging for image segmentation. In *Proceedings of the 22nd Symposium on Information Theory in the Benelux*, pages 189 – 196, 2001.
- [15] J. F. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8:679–700, 1986.
- [16] J. S. Cardoso and L. Corte-Real. Toward a generic evaluation of image segmentation. *IEEE Transactions on Image Processing*, 14:1773–1782, 2005.
- [17] H. Choi and R. G. Baraniuk. Multiscale image segmentation using wavelet-domain hidden markov models. *IEEE Transactions on Image Processing*, 10:1309–1321, 2001.
- [18] W. E. Day. The role of complexity in comparing classifications. *Mathematical Bioscience*, 66:97–114, 1983.
- [19] W. E. Day. Optimal algorithms for comparing trees with labeled leaves. *Journal of Classification*, 2:7–28, 1985.
- [20] M. Van Droogenbroeck and H. Talbot. Segmentation by adaptive prediction and region merging. In *Proceedings of the International Conference on Digital Image Computing: Techniques and Applications*, pages 561–570, 2003.

- [21] D. P. Faith and L. Belbin. Comparison of classifications using measures intermediate between metric dissimilarity and consensus similarity. *Journal of Classification*, 3:257–280, 1986.
- [22] J. S. Farris. On comparing the shapes of taxonomic trees. *Systematic Zoology*, 22:50–54, 1973.
- [23] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59:167–181, 2004.
- [24] E. B. Fowlkes and C. L. Mallows. A method for comparing two hierarchical clusterings. *Journal of the American Statistical Association*, 78:553–569, 1983.
- [25] M. Garland and P. S. Heckbert. Surface simplification using quadric error metrics. In *Proceedings of the International Conference on Computer Graphics and Interactive Techniques*, pages 209–216, 1997.
- [26] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:721–741, 1984.
- [27] The W3C SVG Working Group. Scalable Vector Graphics (SVG) 1.1 Specification. <http://www.w3.org/TR/SVG/>, 2008.
- [28] N. Hansen and A. Ostermeier. Completely derandomized self-adaptation in evolution strategies. *Evolutionary Computation*, 9(2):159–195, 2001.
- [29] L. Hubert and P. Arabie. Comparing partitions. *Journal of Classification*, 2:193–218, 1985.
- [30] E. N. Adams III. Consensus techniques and the comparison of taxonomic trees. *Systematic Zoology*, 19:390–397, 1970.
- [31] X. Jiang, C. Marti, C. Iniger, and H. Bunke. Distance measures for image segmentation evaluation. *EURASIP Journal on Applied Signal Processing*, 2006(1):209–219, 2006.
- [32] S. C. Johnson. Hierarchical clustering schemes. *Psychometrika*, 32(3):241–254, 1967.
- [33] A. Kraskov, H. Stögbauer, R. G. Adrzejak, and P. Grassberger. Hierarchical clustering based on mutual information. <http://arxiv.org/abs/q-bio/0311039>, 2003.

- [34] F.-J. Lapointe and P. Legendre. A statistical framework to test the consensus of two nested classifications. *Systematic Zoology*, 39:1–13, 1990.
- [35] F.-J. Lapointe and P. Legendre. Statistical significance of the matrix correlation coefficient for comparing independent phylogenetic trees. *Systematic Biology*, 41:378–384, 1992.
- [36] Y. V. LeClerc. Constructing simple stable descriptions for image partitioning. *International Journal of Computer Vision*, 3:73–102, 1989.
- [37] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.
- [38] J. Luo and C e G. Perceptual grouping of segmented regions in color images. *Pattern Recognition*, 36:2781–2792, 2003.
- [39] D. T. MacDonald. Image segment processing for analysis and visualization. <http://www.site.uottawa.ca/~dmacd070/ispav>, 2008.
- [40] D. T. MacDonald and J. Lang. Bitmap to vector conversion for multi-level analysis and visualization. In *Proceedings of the International Conference on Scalable Vector Graphics*, 2008.
- [41] D. T. MacDonald, J. Lang, and M. McAllister. Evaluation of colour image segmentation hierarchies. In *Proceedings of the Canadian Conference on Computer and Robot Vision*, 2006.
- [42] T. Margush and F. R. McMorris. Consensus n-trees. *Bulletin of Mathematical Biology*, 43(2):239–244, 1981.
- [43] D. R. Martin, C. C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26:530–549, 2004.
- [44] D. R. Martin, C. C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings of IEEE International Conference on Computer Vision*, volume 2, pages 416–423, July 2001.

- [45] Marina Meilă. Comparing clusterings - an information based approach. *Journal of Multivariate Analysis*, 98:873–895, 2007.
- [46] M. F. Mickevich. Taxonomic congruence. *Systematic Zoology*, 27:143–158, 1978.
- [47] A. Montanvert, P. Meer, and A. Rosenfeld. Hierarchical image analysis using irregular tessellations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(4):307–316, 1991.
- [48] J. M. Morel and S. Solimini. Variational methods in image segmentation. *Progress in Nonlinear Differential Equations and their Applications*, 14, 1995.
- [49] D. Mumford and J. Shah. *Optimal Approximations by Piecewise Smooth Functions and Associated Variational Problems*. Center for Intelligent Control Systems, 1988.
- [50] G. Nelson. Cladistic analysis and synthesis: Principles and definitions, with a historical note on Adanson’s *Familles des Plantes* (1963 - 1964). *Systematic Zoology*, 28:1–21, 1979.
- [51] G. Nelson and N. I. Platnick. *Systematics and Biogeography: Cladistics and Vicariance*. Columbia University Press, New York, NY, 1981.
- [52] P. L. Palmer, H. Dabis, and J. Kittler. A performance measure for boundary detection algorithms. *Computer Vision and Image Understanding*, 63:476–494, 1996.
- [53] D. Parikh and T. Chen. Hierarchical semantics of objects (hSOs). In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1–8, 2007.
- [54] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes: The Art of Scientific Computing, Third Edition*. Cambridge University Press, 2007.
- [55] W. M. Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*, 66:846–850, 1971.
- [56] D. F. Robinson and L. R. Foulds. Comparison of phylogenetic trees. *Mathematical Biosciences*, 53:131–147, 1981.
- [57] P. L. Rosin. Refining region estimates. *International Journal of Pattern Recognition and Artificial Intelligence*, 12:841–866, 1998.

- [58] M. Rotard, K. Otte, D. MacDonald, and C. Taras. SVG4Blind. <http://www.vis.uni-stuttgart.de/~taras/SVG4Blind.html>, 2008.
- [59] P. Salembier and L. Garrido. Binary partition tree as an efficient representation for image processing, segmentation, and information retrieval. *IEEE Transactions on Image Processing*, 9:561–576, 2000.
- [60] R. T. Schuh and J. S. Farris. Methods for investigating taxonomic congruence and their application to the lepto-podomorpha. *Systematic Zoology*, 30:331–351, 1981.
- [61] R. T. Schuh and J. T. Polhemus. Analysis of taxonomic congruence among morphological, ecological, and biogeographic data sets for the lepto-podomorpha (hemiptera). *Systematic Zoology*, 29:1–26, 1980.
- [62] C. E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423, 1948.
- [63] K.-T. Shao and R. R. Sokal. Significance tests of consensus indices. *Systematic Zoology*, 35:582–590, 1986.
- [64] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.
- [65] Y. Shu, G.-A. Bilodeau, and F. Cheriet. Segmentation of laparoscopic images: Integrating graph-based segmentation and multistage region merging. In *Proceedings of the Canadian Conference on Computer and Robot Vision*, pages 429–436, 2005.
- [66] R. R. Sokal and F. J. Rohlf. The comparison of dendrograms by objective methods. *Taxon*, 11:33–40, 1962.
- [67] E. B. Sudderth, A. Torralba, W. T. Freeman, and Alan S. Willsky. Learning hierarchical models of scenes, objects, and parts. In *Proceedings of the IEEE International Conference on Computer Vision*, volume 2, pages 1331–1338, 2005.
- [68] B. Sumengen, B. S. Manjunath, and C. Kenney. Image segmentation using multi-region stability and edge strength. In *Proceedings of the IEEE International Conference on Image Processing*, volume 3, pages 429–432, 2003.
- [69] T. C. Tao and D. J. Crisp. A useful bound for region merging algorithms in a Bayesian model. In *Proceedings of the Australasian Computer Science Conference*, pages 95–100, 2003.

- [70] R. Unnikrishnan and M. Hebert. Measures of similarity. In *Proceedings of the IEEE Workshop on Computer Vision Applications*, pages 394–400, 2005.
- [71] R. Unnikrishnan, C. Pantofaru, and M. Hebert. A measure for objective evaluation of image segmentation algorithms. In *Proceedings of the IEEE Workshop on Empirical Evaluation Methods in Computer Vision*, volume 3, pages 34–41, 2005.
- [72] R. Unnikrishnan, C. Pantofaru, and M. Hebert. Toward objective evaluation of image segmentation algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29, 2007.
- [73] K. L. Vincken, A. S. E. Koster, and M. A. Viergever. Probabilistic multiscale image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:109–120, 1997.
- [74] J. H. Ward. Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Society*, 58:236–244, 1963.
- [75] M. S. Watermand and T. F. Smith. On the similarity of dendrograms. *Journal of Theoretical Biology*, 73:789–800, 1978.
- [76] M. Wilkinson. Common cladistic information and its consensus representation: Reduced Adams and reduced cladistic consensus trees and profiles. *Systematic Biology*, 43:343–368, 1994.
- [77] J. Xuan, T. Adali, and Y. Wang. Segmentation of magnetic resonance brain image: integrating region growing and edge detection. In *Proceedings of the IEEE International Conference on Image Processing*, volume 3, pages 544 – 547, 1995.
- [78] Y. Yitzhaky and E. Peli. A method for objective edge detection evaluation and detector parameter selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25:1027–1033, 2003.
- [79] Y. J. Zhang. A survey on evaluation methods for image segmentation. *Pattern Recognition*, 29:1335–1346, 1996.
- [80] Y. J. Zhang and J. J. Gerbrands. Objective and quantitative segmentation evaluation and comparison. *Signal Processing*, 39:43–54, 1994.

- [81] S. C. Zhu and A. Yuille. Region competition: unifying snakes, region growing, and Bayes/MDL for multiband image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(9):884–900, 1996.
- [82] T. Zöllner and J. M. Buhmann. Robust image segmentation using resampling and shape constraints. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29:1147–1164, 2007.