A General Method for Unsupervised Segmentation of Images Using a Multiscale Approach

Alvin H.Kam and William J.Fitzgerald

Signal Processing Laboratory
University of Cambridge Engineering Department
Trumpington Street, Cambridge CB2 1PZ,
United Kingdom
{ahswk2,wjf}@eng.cam.ac.uk

Abstract. We propose a general unsupervised multiscale approach towards image segmentation. The novelty of our method is based on the following points: firstly, it is general in the sense of being independent of the feature extraction process; secondly, it is unsupervised in that the number of classes is not assumed to be known a priori; thirdly, it is flexible as the decomposition sensitivity can be robustly adjusted to produce segmentations into varying number of classes and fourthly, it is robust through the use of the *mean shift* clustering and Bayesian multiscale processing. Clusters in the joint spatio-feature domain are assumed to be properties of underlying classes, the recovery of which is achieved by the use of the mean shift procedure, a robust non-parametric decomposition method. The subsequent classification procedure consists of Bayesian multiscale processing which models the inherent uncertainty in the joint specification of class and position via a Multiscale Random Field model which forms a Markov Chain in scale. At every scale, the segmentation map and model parameters are determined by sampling from their conditional posterior distributions using Markov Chain Monte Carlo simulations with stochastic relaxation. The method is then applied to perform both colour and texture segmentation. Experimental results show the proposed method performs well even for complicated images.

1 Introduction

The segmentation of an image into an unknown number of distinct and in some way homogeneous regions is a difficult problem and remains a fundamental issue in low-level image analysis. Many different methodologies has been proposed but a process that is highly unsupervised, flexible and robust has yet to be realised.

In this paper, we propose a general unsupervised multiscale approach towards image segmentation. The strength of our method is based on the following points: (i) it is general in the sense of being independent of the feature extraction process; consequently, the algorithm can be applied to perform different types of segmentation without modification, be it grey-scale, texture, colour based etc.

D. Vernon (Ed.): ECCV 2000, LNCS 1843, pp. 69–84, 2000.

(ii) it is unsupervised in that the number of classes is not assumed to be known a priori (iii) it is flexible as the decomposition sensitivity can be robustly adjusted to produce segmentations into varying number of classes (iv) it is robust through the use of the *mean shift* clustering and Bayesian multiscale processing (v) dramatic speed-ups of computation can be achieved using appropriate processor architecture as most parts of the algorithm are highly parallellised.

The complete algorithm consists of a two-step strategy. Firstly, salient features which correspond to clusters in the feature domain, are regarded as manifestations of classes, the recovery of which is to be achieved using the mean shift procedure [5], a kernel-based decomposition method, which can be shown to be the generalised version of the k-means clustering algorithm [3].

Secondly, upon determining the number of classes and the properties of each class, we proceed towards the problem of classification. Unfortunately, classification in the image segmentation context is afflicted by uncertainties which render most simple techniques ineffective. To be more certain of the class of a pixel requires averaging over a larger area, which unfortunately makes the location of the boundary less certain. In other words, localisation in class space conflicts directly with the simulteneous localisation in position space. This has been rigorously shown by Wilson and Spann [15] to be a consequence of the relationship between the signals of which images are composed and the symbolic descriptions, in terms of classes and properties, which are the output of the segmentation process. These effects of uncertainties can however be minimised by the use of representations employing multiple scales.

Motivated by this rationale, we adopted a Bayesian multiscale classification paradigm by modelling the inherent uncertainty in the joint specification of class and position via the Multiscale Random Field model [1]. This approach provides context for the classification at coarser scales before achieving accurate boundary tracking at finer resolutions.

2 The Mean Shift Procedure

The mapping of real images to feature spaces often produces a very complex structure. Salient features whose recovery is necessary for the solution of the segmentation task, correspond to clusters in this space. As no a priori information is typically available, the number of clusters/classes and their shapes/distributions have to be discerned from the given image data.

The uniqueness of image analysis in this clustering context lies in the fact that features of neighbouring data points in the spatial domain are strongly correlated. This is due to the fact that typical images do not consist of random points but are manisfestations of entities which form contiguous regions in space. Following this rationale, we represent the image to be segmented in a n-dimensional feature space. Position and feature vectors are then concatenated to obtain a joint spatio-feature domain of dimension $d = n + 2$. Our approach thus includes the crucial spatial locality information typically missing from most

clustering approaches to image segmentation. All features are then normalised by dividing with its standard deviation to eliminate bias due to scaling.

This joint spatio-feature domain can be regarded as samples drawn from an unknown probability distribution function. If the distribution is represented with a parametric model (e.g. Gaussian mixture), severe artifacts may be introduced as the shape of delineated clusters is constrained. Non-parametric cluster analysis however, uses the modes of the underlying probability density to define cluster centres and the valleys in the density to define boundaries separating the clusters.

Kernel estimation is a good practical choice for non-parametric clustering techniques as it is simple and for kernels obeying mild conditions, the estimation is asymptotically unbiased, consistent in a mean-square sense and uniformly consistent in probability [5]. Furthermore, for unsupervised segmentation, where flexibility and interpretation are of utmost importance, any rigid inference of 'optimal' number of clusters may not be productive. By using a kernel-based density estimation approach and controlling the kernel size, a method is developed which is capable of decomposing an image into the number of classes which corresponds well to a useful partitioning for the application at hand. Alternatively, we can produce a set of segmentations for the image (corresponding to different number of classes) with each one reflecting the decomposition of the image under different feature resolution.

2.1 Density Gradient and the Mean Shift Vector

Let $\{\mathbf{X}_i\}_{i=1\ldots N}$ be the set of N image vectors in the d-dimensional Euclidean space R^d. The multivariate kernel density estimate obtained with kernel $K(\mathbf{x})$ and window radius h, computed at point $\mathbf{x}$ is defined as:

$$\hat{f}(\mathbf{x}) = \frac{1}{Nh^d} \sum_{i=1}^{N} K\left(\frac{\mathbf{x} - \mathbf{X}_i}{h}\right) \tag{1}$$

The use of a differential kernel allows us to define the estimate of the density gradient estimate as the gradient of the kernel density estimate (1):

$$\hat{\nabla} f(\mathbf{x}) \equiv \nabla \hat{f}(\mathbf{x}) = \frac{1}{Nh^d} \sum_{i=1}^{N} \nabla K\left(\frac{\mathbf{x} - \mathbf{X}_i}{h}\right) \tag{2}$$

The Epanechnikov kernel [13], given by:

$$K_{\mathrm{E}}(\mathbf{x}) = \begin{cases} \frac{1}{2} c_{\mathrm{d}}^{-1} (d+2)(1 - \mathbf{x}^T \mathbf{x}) \text{ if } \mathbf{x}^T \mathbf{x} < 1 \\ 0 \qquad\qquad\qquad\qquad \text{otherwise} \end{cases} \tag{3}$$

has been shown to be the simplest kernel to possess properties of asymptotic unbiasedness, mean-square and uniform consistency for the density gradient estimate [5]. In this case, the density gradient estimate becomes:

$$\hat{\nabla} f_{\mathrm{E}}(\mathbf{x}) = \frac{N_{\mathbf{x}}}{N(h^d c_{\mathrm{d}})} \frac{d+2}{h^2} \left[\frac{1}{N_{\mathbf{x}}} \sum_{\mathbf{X}_i \in S_{\mathrm{h}}(\mathbf{x})} (\mathbf{X}_i - \mathbf{x}) \right] \tag{4}$$

where the region $S_{\mathrm{h}}(\mathbf{x})$ is a hypersphere (uniform kernel) of radius h centred on $\mathbf{x}$, having the volume $h^d c_{\mathrm{d}}$ and containing $N_{\mathbf{x}}$ data points. The last term in (4):

$$M_{\mathrm{h}}(\mathbf{x}) = \frac{1}{N_{\mathbf{x}}} \sum_{\mathbf{X}_i \in S_{\mathrm{h}}(\mathbf{x})} (\mathbf{X}_i - \mathbf{x}) \tag{5}$$

is called the sample *mean shift*. The quantity $\frac{N_{\mathbf{x}}}{N(h^d c_{\mathrm{d}})}$ is the kernel density estimate computed with the uniform kernel $S_{\mathrm{h}}(\mathbf{x})$, $\hat{f}_{\mathrm{U}}(\mathbf{x})$ and thus we can write (4) as:

$$\hat{\nabla} f_{\mathrm{E}}(\mathbf{x}) = \hat{f}_{\mathrm{U}}(\mathbf{x}) \frac{d+2}{h^2} M_{\mathrm{h}}(\mathbf{x}) \tag{6}$$

which yields:

$$M_{\mathrm{h}}(\mathbf{x}) = \frac{h^2}{d+2} \frac{\hat{\nabla} f_{\mathrm{E}}(\mathbf{x})}{\hat{f}_{\mathrm{U}}(\mathbf{x})} \tag{7}$$

Equation (7) depicts the mean shift vector as a normalised density gradient estimate. This implies that the vector always points towards the direction of the maximum increase in density and hence it can define a path leading to a local density maximum. The normalised gradient in (7) also brings about a desirable adaptive behaviour, with the mean shift step being large for low density regions and decreases as $\mathbf{x}$ approaches a mode.

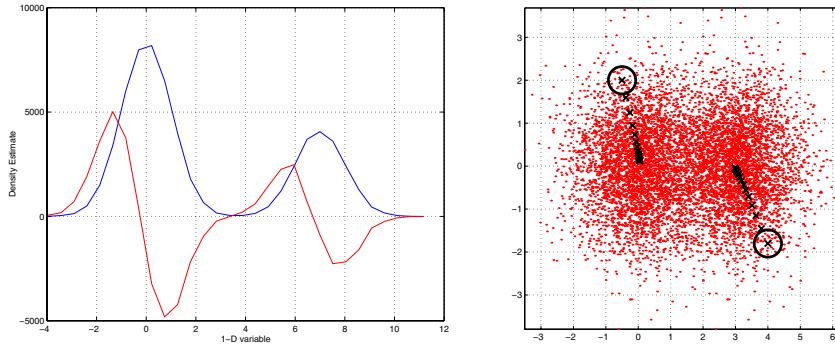

Fig. 1. On the left: Consider the density estimation plot (in blue) of a hypothetical 1-D feature. The gradient or derivative of the density plot is shown in red. It is obvious that the density gradient always points in the direction of maximum increase in density (bear in mind that left-to-right along the 1-D axis constitutes positive movement). On the right: As the mean shift vector is proportional to the density gradient estimate, successive computations of the mean shift define a path leading to a local density maximum (shown here for a 2-D feature)

While it is true that the mean shift vector $M_{\mathrm{h}}(\mathbf{x})$ has the direction of the gradient estimate at $\mathbf{x}$, it is not apparent that the density estimate at the suc-

cessive locations of the mean shift procedure is a monotonic increasing sequence. The following theorem, however, assures the convergence:

Theorem. Let $\hat{f}_{\mathrm{E}} = \left\{ \hat{f}_k \left(\mathbf{Y}_k, K_{\mathrm{E}} \right) \right\}_{k=1,2,\ldots}$ be the sequence of density estimates obtained using the Epanechnikov kernel and computed at the points $(\mathbf{Y}_k)_{k=1,2,\ldots}$ defined by the sucessive locations of the mean shift procedure with a uniform kernel. The sequence is convergent.

Proof of this theorem can be found in [4].

2.2 Mean Shift Clustering Algorithm

The mean shift clustering algorithm consists of successive computation of the mean shift vector, $M_{\mathrm{h}}(\mathbf{x})$ and translation of the window $S_{\mathrm{h}}(\mathbf{x})$ by $M_{\mathrm{h}}(\mathbf{x})$. Each data point thus becomes associated with a point of convergence which represents a local mode of the density in the d-dimensional space. Iterations of the procedure thus gives rise to a 'natural' clustering of the image data, based solely on their mean shift trajectories.

The procedure in its original form, is meant to be applied to each point in the data set. This approach is not desirable for practical applications especially when the data set is large as is typical for images. The conventional mean shift procedure has a complexity of $O(N^2)$ for a set of N data points. A more realistic approach consist of a probabilistic mean shift algorithm as proposed in [4] whose complexity is of $O(mN)$, with $m \ll N$, as outlined below:

1. *Define a random tessellation of the space with $m \ll N$ hyperspheres $S_{\mathrm{h}}(\mathbf{x})$.* To reduce computational load, a set of m points called the sample set, is randomly selected from the data. It is proposed that two simple constraints are imposed on the sample set: firstly, the distance between any two points in the sample set should not be smaller than h, the radius of the hypersphere, $S_{\mathrm{h}}(\mathbf{x})$. Secondly, sample points should not lie in sparsely populated regions. A region is defined as sparsely populated whenever the number of points inside the hypersphere is below a certain threshold T_1. The distance and density constraints automatically determine the size m of the sample set. Hyperspheres centred on the sample set cover most of the data points. These constraints can of course be relaxed if processing time is not a critical issue.
2. *The mean shift procedure is applied to the sample set.* A set containing m cluster centre candidates is defined by the points of convergence of the m mean shift procedures. As the computation of the mean shift vectors is based on almost the entire data set, the quality of the gradient estimate is not diminished by the use of sampling.
3. *Perturb the cluster candidates and reapply the mean shift procedure.* Since a local plateau can prematurely stop the iterations, each cluster centre candidate is perturbed by a random vector of small norm and the mean shift procedure is left to converge again.

4. *Derive the cluster centres* $\mathbf{Y}_1, \mathbf{Y}_2, ..., \mathbf{Y}_p$ *from the cluster centre candidates.* Any subset of cluster centre candidates which are less than distance h from each other defines a cluster centre. The cluster centre is the mean of the cluster centre candidates in the subset.
5. *Validate the cluster centres.* Between any two cluster centres $\mathbf{Y}_i$ and $\mathbf{Y}_j$, a significant valley should occur in the underlying density. The existence of the valley is tested for each pair $(\mathbf{Y}_i, \mathbf{Y}_j)$. The hypersphere $S_{\mathrm{h}}(\mathbf{x})$ is moved with step h along the line defined by $(\mathbf{Y}_i, \mathbf{Y}_j)$ and the density is estimated using the Epanechnikov kernel, K_{E} along the line. Whenever the ratio between $\min\left[\hat{f}(\mathbf{Y}_i), \hat{f}(\mathbf{Y}_j)\right]$ and the minimum density along the line is larger than a certain threshold, T_2, a valley is assumed between $\mathbf{Y}_i$ and $\mathbf{Y}_j$. If no valleys are found, the cluster centre of lower density, ($\mathbf{Y}_i$ or $\mathbf{Y}_j$) is removed from the set of cluster centres.

The clustering algorithm makes use of three parameters: the kernel radius, h, which controls the sensitivity of the decomposition, the threshold T_1, which imposes the density constraint on the sample set and T_2, corresponding to the minimum acceptable peak-valley ratio. The parameters T_1 and T_2 generally have a weak influence on the final results. In fact, all our experimental results as performed on 256×256 resolution images were obtained by fixing $T_1 = 50$ and $T_2 = 1.2$. As the final objective of a segmentation is often application specific, top-down a priori information controls the kernel radius h, resulting in data points having trajectories that merge into appropriate number of classes. Alternatively, the 'optimal' radius can be obtained as the centre of the largest operating range which yields the same number of classes. Finally, cluster centres which are sufficient close (distance being less then h apart) in the n-dimensional 'feature-only' space (remember, $n = d - 2$) are merged in order to group similar features which are spatially distributed.

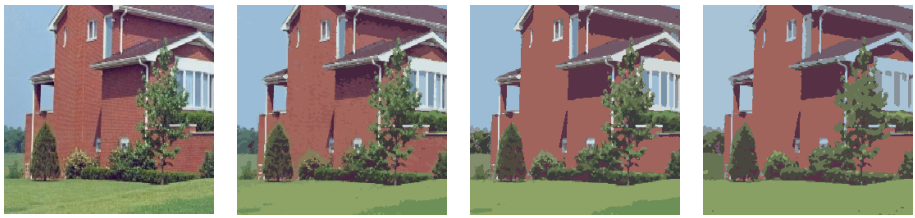

Fig. 2. Flexibility of mean shift clustering in determining the number of classes. From left: Image of 'house' and its corresponding segmentations using $h = 0.2$ (47 classes), $h = 0.4$ (15 classes) and $h = 0.8$ (8 classes) in the 5-dimensional normalised Euclidean space. The classification strategy is implemented using techniques detailed in Sect. 3 and 4

We shall assume these validated cluster centres to be manisfestations of underlying class properties for our image segmentation task, with each class thus

represented by an n-dimensional feature vector. We then proceed with a multiscale Bayesian classification algorithm outlined below. A Bayesian approach is used because the notion of likelihood can be determined naturally from the computation of dissimilarity measures between feature vectors. Moreover, priors can be effectively used to represent information regarding segmentation results of coarser scales when segmentation is being performed for finer resolutions.

3 The Multiscale Random Field Model

A multiscale Bayesian classification approach is implemented using the Multiscale Random Field (MSRF) model [1]. In this model, let the random field Y be the image that must be segmented into regions of distinct statistical behaviour. The behaviour of each observed pixel is dependent on a corresponding unobserved class in X. The dependence of observed pixels on their class is specified through the probability $p(Y = y|X = x)$, or the likelihood function. Prior knowledge about the size and shapes of regions will be modelled by the prior distribution $p(X)$.

X is modelled by a pyramid structure multiscale random field. $X^{(0)}$ is assumed to be the finest scale random field with each site corresponding to a single image pixel. Each site at the next coarser scale, $X^{(1)}$, corresponds to a group of four sites in $X^{(0)}$. And the same goes for coarser scales upwards. Thus, the multiscale classification is denoted by the set of random fields, $X^{(n)}, n = 0, 1, 2, ...$

The main assumption made is that the random fields form a Markov Chain from coarse to fine scale, that is:

$$p\left(X^{(n)} = x^{(n)}|X^{(l)} = x^{(l)}, l > n\right) = p\left(X^{(n)} = x^{(n)}|X^{(n+1)} = x^{(n+1)}\right) \quad (8)$$

In other words, it is assumed that for $X^{(n)}$, $X^{(n+1)}$ contain all relevant information from previous coarser scales. We shall further assume that the classification of sites at a particular scale is dependent only on the classfication of a local neighbourhood at the next coarser scale. This relationship and the chosen neighbourhood structure are depicted in Fig. 3.

3.1 Sequential Maximum a Posteriori (SMAP) Estimation

In order to segment the image Y, one must accurately estimate the site classes in X. Generally, Bayesian estimators attempt to minimise the average cost of an errorneous segmentation. This is done by solving the optimisation problem:

$$\hat{x} = \arg\min_{x} E\left(C(X, x)|Y = \mathbf{y}\right) \quad (9)$$

where $C(X, x)$ is the cost of estimating the 'true' segmentation X by the approximate segmentation x. The choice of functional C is of crucial importance as it determines the relative importance of errors. Ideally, a desirable cost function should assign progressively greater cost to segmentations with larger regions of

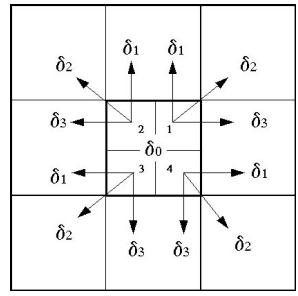

Fig. 3. Blocks 1,2,3 and 4 are of scale n and have a common parent at scale $n+1$, i.e. δ_0, which they are dependent on. The arrows show additional dependence on their parent's neighbours: δ_1, δ_2 and δ_3

misclassified pixels. To achieve this goal, the following cost function has been proposed [1]:

$$C_{\text{SMAP}} = \frac{1}{2} + \sum_{n=0}^{L} 2^{n-1} C_n(X, x) \tag{10}$$

where:

$$C_n(X, x) = 1 - \prod_{i=n}^{L} \delta\left(X^{(i)} - x^{(i)}\right) \tag{11}$$

The behaviour of C_{SMAP} is solely a function of the coarsest scale that contains a misclassified site. The solution is given by:

$$\hat{x}^{(n)} = \arg\max_{x^{(n)}} \left\{ p\left(X^{(n)} = x^{(n)} | X^{(n+1)} = \hat{x}^{(n+1)}, Y = \mathbf{y}\right) + \varepsilon\left(x^{(n)}\right) \right\} \tag{12}$$

where ε is a second order term which may be bounded by:

$$0 \leq \varepsilon(x^{(n)}) \leq \max_{x^{(n-1)}} p\left(X^{(n-1)} = x^{(n-1)} | X^{(n)} = x^{(n)}, Y = \mathbf{y}\right) << 1 \tag{13}$$

Using Bayes rule and ignoring the contribution of ε, one obtains the following equation:

$$\hat{x}^{(n)} = \begin{cases} \arg\max\limits_{x^{(L)}} \left\{ p\left(Y = \mathbf{y} | X^{(L)} = x^{(L)}\right) p(X^{(L)} = x^{(L)}) \right\} & \text{for } n = L \\ \arg\max\limits_{x^{(n)}} \left\{ p\left(Y = \mathbf{y} | X^{(n)} = x^{(n)}\right) p(X^{(n)} = x^{(n)} | X^{(n+1)} = \hat{x}^{(n+1)}) \right\} & \text{for } n < L \end{cases} \tag{14}$$

where L is the coarsest scale of the multiscale pyramid. The solution is initialised by determining the maximum a posteriori (MAP) estimate of the coarsest scale field given the image Y. The MAP segmentation at the next finer scale, $\hat{x}^{(n)}$ is then found by computing the MAP estimate of $X^{(n)}$, given $\hat{x}^{(n+1)}$ and the image Y, hence the name sequential MAP (SMAP) estimator. For our experiments, we assumed a uniform prior for $X^{(L)}$ but in general, any suitable priors may be used.

3.2 Likelihood and Prior Probability Functions

We will assume that at a particular scale, the observed sites are *conditionally* independent given their classes:

$$p\left(Y = \mathbf{y}|X^{(n)} = x^{(n)}\right) = \prod_{s \in S^{(n)}} p\left(Y_s = \mathbf{y}_s|X_s^{(n)} = x_s^{(n)}\right) \tag{15}$$

where the index s denotes individual sites at scale n, $\mathbf{y}_s$ represents the 'averaged' feature vector of observed site Y_s and $x_s^{(n)}$ correspond to segmentation classes which have values taken from $\Lambda = \{1, 2, ..., c\}$, where c is the total number of classes.

The multiscale averaging to generate $\mathbf{y}_s$ at each scale is achieved using the lowpass subimages of Kingsbury's complex wavelet decomposition (KCWD) [10] of each feature component of $\mathbf{y}$. The advantage of KCWD over the more conventional discrete wavelet transform for multiscale representation of features lies in the remarkable shift invariance property of the former approach. To illustrate, the figure below shows grey-level feature averaging of 'lenna' using the lowpass subimages of KCWD:

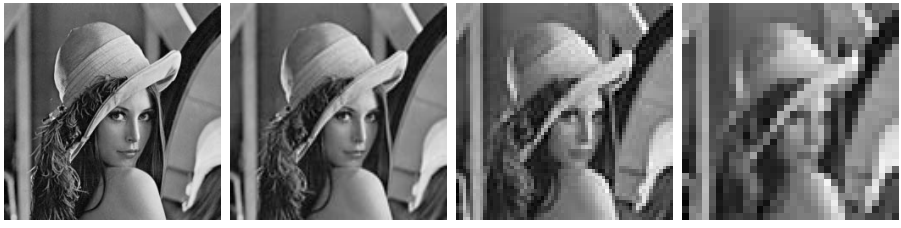

Fig. 4. Grey-level feature averaging of 'lenna' using the lowpass subimages of Kingsbury's complex wavelet decompostion at scales (from left) $n = 0, 1, 2$ and 3 respectively, with excellent shift invariance

We choose to model $p(Y_s = \mathbf{y}_s|X_s^{(n)} = x_s^{(n)})$ as a Gaussian distribution:

$$p\left(Y_s = \mathbf{y}_s|X_s^{(n)} = x_s^{(n)}\right) \propto \frac{1}{\sigma_n} \exp\left\{-\frac{1}{2\sigma_n^2}\left\|\mathbf{y}_s, x_s^{(n)}\right\|^2\right\} \tag{16}$$

where $\|\cdot\|$ denotes Euclidean distance. The variance parameter σ_n typically increases with segmentation resolution, which agrees with the increased class uncertainties at finer scales.

From our assumptions on the label field X, we have:

$$p\left(X^{(n)} = x^{(n)}|X^{(n+1)} = \hat{x}^{(n+1)}\right) = \prod_{s \in S^{(n)}} p\left(X_s^{(n)} = x_s^{(n)}|X_{\delta s}^{(n+1)} = \hat{x}_{\delta s}^{(n+1)}\right) \tag{17}$$

where δs denotes the neighbourhood structure shown in Fig. 3. We choose the following form for the right-hand-side term above:

$$p\left(X_s^{(n)} = m | X_{\delta s_0}^{(n+1)} = i, X_{\delta s_1}^{(n+1)} = j, X_{\delta s_2}^{(n+1)} = k, X_{\delta s_3}^{(n+1)} = l\right) = \frac{\alpha_n}{9}\left(3\delta_{m,i} + 2\delta_{m,j} + 2\delta_{m,k} + 2\delta_{m,l}\right) + \frac{1-\alpha_n}{c} \quad (18)$$

where $\delta_{m,n}$ represents the unit delta function. The scale dependent parameter $\alpha_n \in [0,1]$, determines the probability that the class of the fine scale site remains the same as that of one of the coarser scale local neighbourhood. Conversely, $1-\alpha_n$ is the probability that a new class will be randomly chosen from the remaining classes.

3.3 Parameter Estimation

In order for the method to be adaptive to the segmentation at hand, the MSRF model parameters has to be estimated at each scale. A Markov Chain Monte Carlo (MCMC) sampling approach is used in a predetermined sequential scan to sample the model parameters and the segmentation map from their conditional distributions in a specific order. The conditional distributions of the segmentation map and the model parameters are difficult functions to maximise because they are multimodal and the vast combined parameter spaces are composed of both continuous and discrete subspaces. The Metropolis-Hastings algorithm [7], [11] is a robust MCMC optimisation algorithm which is ideally suited to be applied to these types of problem.

The stochastic relaxation process of simulated annealing [6] is used. At initial high temperatures, the probability of acceptance is very high but it reduces with the gradual cooling of the annealing temperature to reach the global maximum at very low temperatures. The first step consist of sampling the class field. The conditional distribution, from equations (15) and (17), is given by:

$$p\left(X^{(n)} = x^{(n)} | X^{(n+1)} = \hat{x}^{(n+1)}, Y = \mathbf{y}, \sigma_n, \alpha_n\right) \propto \left\{ \prod_{s \in S^{(n)}} \left[p(Y_s = \mathbf{y}_s | X_s^{(n)} = x_s^{(n)}, \sigma_n) p(X_s^{(n)} = x_s^{(n)} | X_{\delta s}^{(n+1)} = \hat{x}_{\delta s}^{(n+1)}, \alpha_n) \right] \right\}^{\frac{1}{T_t}} \quad (19)$$

where T_t is the annealing temperature at iteration t of the algorithm and the distributions for the likelihood and prior terms are given by (16) and (18) respectively.

For the sampling of σ_n and α_n, the respective conditional distributions are:

$$\sigma_n : p(\sigma_n | X^{(n)} = x^{(n)}, Y = \mathbf{y}) \propto \left\{ \left[\prod_{s \in S^{(n)}} p(Y_s = \mathbf{y}_s | X_s^{(n)} = x_s^{(n)}, \sigma_n) \right] p(\sigma_n | X_s^{(n)} = x_s^{(n)}) \right\}^{\frac{1}{T_t}} \quad (20)$$

$$\alpha_n : p(\alpha_n | X^{(n)} = x^{(n)}, X^{(n+1)} = \hat{x}^{(n+1)}) \propto$$
$$\left\{ \left[\prod_{s \in S^{(n)}} p(X_s^{(n)} = x_s^{(n)} | X_{\delta s}^{(n+1)} = \hat{x}_{\delta s}^{(n+1)}, \alpha_n) \right] p(\alpha_n | X_{\delta s}^{(n+1)} = \hat{x}_{\delta s}^{(n+1)}) \right\}^{\frac{1}{T_t}} \tag{21}$$

with the likelihood terms given by equations (16) and (18) for σ_n and α_n respectively. Non-informative or reference priors [9] were used for all experiments.

Our choice of the likelihood and prior probability distributions also makes it possible for the dissimilarity term of (16) and the delta function terms of (18) to be calculated for each segmentation class prior to the MCMC sampling procedure. Therefore, these terms need to be computed only once and not repeatedly for each iteration of the Metropolis-Hastings algorithm. This greatly decreases the overall computation time. More importantly, as the computation of conditional distributions at each site is independent of each other at a particular iteration, dramatic speed-ups of calculations can be achieved using systems with highly parallel architecture.

There has been much debate of how convergence might relate to the annealing schedule used. Theoretically, the logarithmic schedule of [6] is guaranteed to converge in infinite time. In practice, this is not implementable. We have adopted a linear schedule which produces robust convergence in a relatively short time.

We now apply the complete algorithm to perform the challenging tasks of **colour** and **texture** segmentation.

4 Colour Segmentation

Colour correlates with the class identity of an object because pigments form part of the appearance of an object and thus provide vital cues for segmentation purposes. In our paper, the perceptually uniform CIE L*a*b* space is used to represent colour features. It is generated by linearly transforming the RGB colour space to the XYZ colour space followed by a non-linear transformation. The non-linear transformation is determined by relation to a nominally white object-colour stimulus which gives the tristimulus values (X_n,Y_n, Z_n). The lightness L^* is given by:

$$L^* = \begin{cases} 116(Y/Y_n)^{\frac{1}{3}} - 16 & \text{for } (Y/Y_n) > 0.008856 \\ 903.3(Y/Y_n) & \text{for } (Y/Y_n) \leq 0.008856 \end{cases} \tag{22}$$

The values a^*,b^* are given as follows:

$$a^* = 500 \{f(X/X_n) - f(Y/Y_n)\} \tag{23}$$

$$b^* = 200 \{f(Y/Y_n) - f(Z/Z_n)\} \tag{24}$$

where:

$$f(t) = \begin{cases} t^{\frac{1}{3}} & \text{for } t > 0.008856 \\ 7.787t + \frac{16}{116} & \text{for } t \leq 0.008856 \end{cases} \tag{25}$$

The distance between two colours as evaluated in L*a*b* space is simply the Euclidean distance between them:

$$\Delta \mathrm{E}_{\mathrm{L^*a^*b^*}} = \sqrt{(\Delta \mathrm{L}^*)^2 + (\Delta \mathrm{a}^*)^2 + (\Delta \mathrm{b}^*)^2} \tag{26}$$

The CIE L*a*b* is as close to be perceptually linear as any colour space is expected to get. Thus the distance measure in (26) effectively quantifies the perceived difference between colours.

Figures 5 and 6 show some typical colour segmentation results using our algorithm. To determine the number of classes, mean shift clustering using $h = 0.7$ (in the normalised 5-dimensional Euclidean space) were used for all experiments to demonstrate that the kernel radius h is a robust parameter that does not require tedious 'trial-and-error' tinkering to achieve desired results for each image.

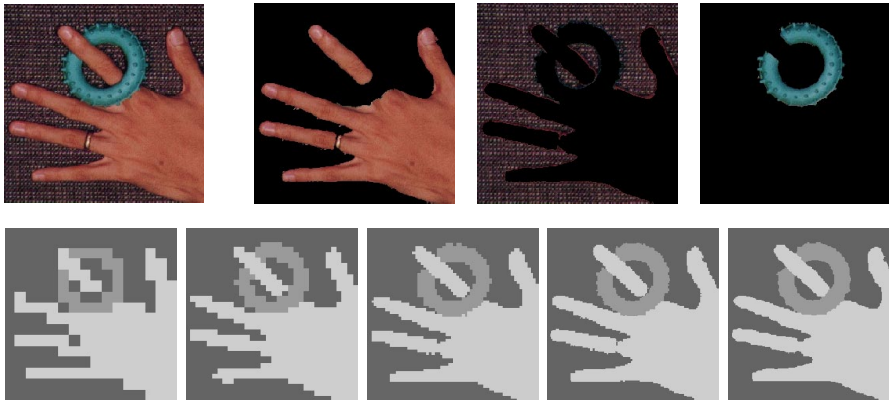

Fig. 5. First row: The 'hand' image and the three classes segmented by the algorithm. Second row: Segmentation results shown at every intermediate scale corresponding to (from left) $n = 4, 3, 2, 1$ and 0 respectively

Segmentation of the 'hand' image shown in figure 5 shows the algorithm being able to easily distinguish the human hand and the blue doughnut-like object from the textured background. As shown by the segmentation result at each scale, processing at coarse scales gives context to the segmentation based on which processing at finer resolution achieves boundary refinement accuracy.

Figure 6 shows more colour segmentation results. For the 'jet' image, the toy jet-plane, its shadow and the background are picked out by the algorithm despite

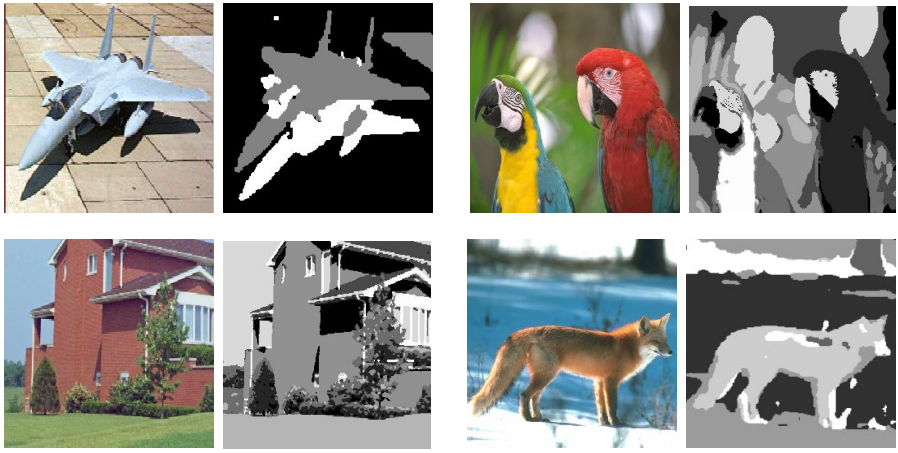

Fig. 6. First row: The 'jet' and 'parrot' image and their corresponding segmentations. Second row: The 'house' and 'fox' image and their corresponding segmentations

the considerable colour variability of each object. Segmentation of the 'parrot' image reveals a fairly smooth partitioning with all major colours bounded by reasonably accurate boundaries. The algorithm also produces a meaningful segmentation of the 'house' image with the sky, walls, window frames, lawn and trees/hedges isolated as separate entities. The 'fox' image poses a tricky problem with its shadows and highlights but the algorithm still performs reasonably well in isolating the fox from the background although there is inevitable misclasification at the extreme light and dark regions of the fox due the L*a*b* features used. Generally, for all the images, the well-defined region contours reflect the excellent boundary tracking ablility of the algorithm while smooth regions of homogeneous behaviour are the result of the multiscale processing.

5 Texture Segmentation

The figure below [12] illustrates a texture feature extraction model. Basically $x(m,n)$ is the input texture image which is filtered by $h(k,l)$, a frequency and orientation selective filter, the output of which passes a local energy function (consisting of a non-linear operator, $f(.)$ and a smoothing operator, $w(k,l)$) to produce the final feature image, $v(m,n)$. Basically, the purpose of the filter, $h(k,l)$, is extraction of spatial frequencies (of a particular scale and orientation) where one or more textures have high signal energy and the others have low energy. A quadrature mirror wavelet filter bank, used in an undecimated version of an adaptive tree-structured decomposition scheme [2], perform this task for our experiments on textures.

Numerous non-linearity operators, $f(.)$, have been applied in the literature, the most popular being the magnitude, $|x|$, the squaring, $(x)^2$ and the rectified

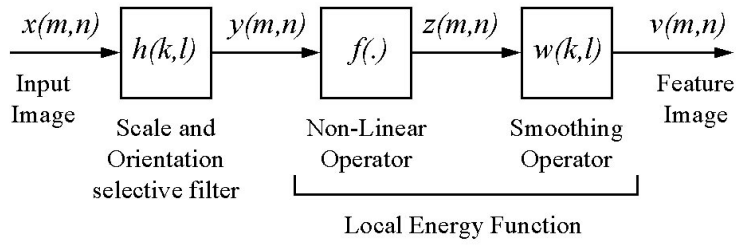

Fig. 7. Block diagram of the texture feature extraction model

sigmoid, $|\tanh(\alpha x)|$. It has been found that squaring in conjuction with the logarithm after the smoothing to be the best operator pair for unsupervised segmentation from a set of tested operator pairs [14]. For this reason, this operator pair is used for our experiments.

Several smoothing filters are possible for $w(k,l)$ and the Gaussian lowpass filter is one candidate. The Gaussian lowpass filter has joint optimum resolution in the spatial and spatial frequency domains, with its impulse response given by:

$$w_{\mathrm{G}}(k,l) = \frac{1}{2\pi\sigma_{\mathrm{s}}^2} \exp\left\{-\frac{(k^2+l^2)}{2\sigma_{\mathrm{s}}^2}\right\} \tag{27}$$

If we want to estimate the local energy of a signal with low spatial frequency, the smoothing filter must have a larger region-of-support and vice versa. Hence, the smoothing filter size may be set to be a function of the band centre frequency, f_0. With f_0 normalised ($-\frac{1}{2} \leq f_0 \leq \frac{1}{2}$), it has been suggested [8] that:

$$\sigma_{\mathrm{s}} = \frac{1}{2\sqrt{2}\,|f_0|} \tag{28}$$

This smoothing filter is also scaled so as to produce unity gain in order for the mean of the filter's output to be identical to that of its input.

For dimension reduction and extraction of saliency, *principal component analysis* is performed on the raw wavelet features, $v(m,n)$. The final feature space for the texture segmentation task consists of two dimensions of textural features (the top two principal components, which typically contribute more than 85% of the total variances of the wavelet features) and one dimension of luminance.

Figure 8 shows some texture segmentation results. Again, as in colour segmentation, mean shift clustering with kernel radius $h = 0.7$ is used to determine the number of classes. For the 'brodatz' image, the algorithm is able to distinguish all 5 textures of the Brodatz texture mosaic and produced a highly accurate segmentation map. The segmentation of the SAR image, 'sar' depicts remarkable preservation of details as well as accurate boundary detection. The image 'manassas', an aerial view of the city of Manassas, Virginia provides an interesting challenge to the algorithm, which as shown, is able to successfully isolate

densely populated areas from roads and flat plains. The leopard of the image 'leo' is also successfully segmented from background grass and scrubs; 'misclassified' regions constitute shadows and relatively large homogeneous regions of black spots on the legs.

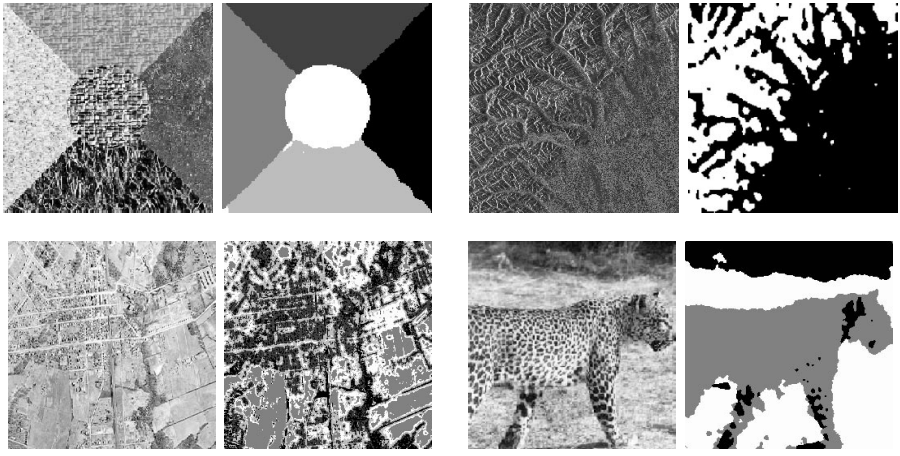

Fig. 8. First row: The 'brodatz' and 'sar' image and their corresponding segmentations. Second row: The 'manassas' and 'leo' image and their corresponding segmentations

6 Summary and Discussion

In this paper, we have proposed a general multiscale approach for unsupervised image segmentation. The method is general due to its independence of the feature extraction process and unsupervised in that the number of classes is not known a priori. The algorithm is also highly flexible due to its ability to control segmentation sensitivity and robust through the use of the mean shift procedure and multiscale processing.

The mean shift procedure has been proven to perform well in detecting clusters of complicated feature spaces of many real images. By controlling the kernel size, the procedure is capable of producing classes whose associative properties correspond well to a meaningful partitioning of an image. The Multiscale Random Field model makes effective use of the inherent trade-off between class and position uncertainty which is evident through the excellent boundary tracking performance. This multiscale processing reduces computational costs by keeping computations local and yet produces results that reflect the global properties of the image.

The proposed method has been shown to perform well for colour and texture segmentation of various images. It produces desirable segmentations with smooth regions of homogeneous behaviour and accurate boundaries. We believe these

segmentations possess a high degree of utility especially as precursors to higher level tasks of scene analysis or object recognition.

References

1. Bouman, C. , Shapiro, M.: A Multiscale Random Field Model for Bayesian Image Segmentation. IEEE Trans. Image Process. **3**(2) (1994) 162–177
2. Chang, T. , Kuo, C.J.: Texture Analysis and Classification with a Tree-Structured Wavelet Transform. IEEE Trans. Image Process. **2**(4) (1993) 429–441
3. Cheng, Y.: Mean Shift, Mode Seeking, and Clustering. IEEE Trans. Pattern Anal. Machine Intell. **17**(8) (1993) 770–799
4. Comaniciu, D. , Meer, P.: Distribution Free Decomposition of Multivariate Data. 2nd Intern. Workshop on Statist. Techniques in Patt. Recog., Sydney, Australia. (1998)
5. Fukunaga, K. , Hosteler, L.D.: The Estimation of the Gradient of a Density Function, with Applications in Pattern Recognition. IEEE Trans. Info. Theory **21** (1975) 32–40
6. Geman, S. , Geman, D.: Stochastic Relaxation, Gibbs Distributions and the Bayesian Restoration of Images. IEEE Trans. Pattern Anal. Machine Intell. **6**(6) (1984) 721–741
7. Hastings, W.K.: Monte Carlo Sampling Methods using Markov Chains and their Applications. Biometrika **57** (1970) 97–109
8. Jain, A.K. , Farrokhnia, F.: Unsupervised Texture Segmentation using Gabor Filters. Pattern Recognition. **24**(12) (1991) 1167–1186
9. Jeffreys, H.: Theory of Probability. Oxford University Press (1939)
10. Kingsbury, N.: The Dual-Tree Complex Wavelet Transform: A New Technique for Shift Invariance and Directional Filters. IEEE Dig. Sig. Proc. Workshop, DSP98, Bryce Canyon, paper no. 86. (1998)
11. Metropolis, N. , Rosenbluth, A.W. , Rosenbluth, M.N.: Equation of State Calculations by Fast Computing Machines. Journal of Chem. Phys. **21** (1953) 1087–1092
12. Rangen, T. , Husoy, J.H.: Multichannel Filtering for Image Texture Segmentation. Opt. Eng. **8** (1994) 2617–2625
13. Silverman, B.W.: Density Estimation for Statistics and Data Analysis. Chapman and Hall, London. (1986)
14. Unser, M. , Eden, M.: Non-linear Operators for Improving Texture Segmentation based on Features Extracted by Spatial Filtering. IEEE Trans. Syst., Man and Cyb. **20** (1990) 804–815
15. Wilson, R. , Spann, M.: Image Segmentation and Uncertainty. Research Studies Press Ltd., Letchworth, Hertfordshire, U.K. (1988)