MEAN SHIFT SPECTRAL CLUSTERING FOR PERCEPTUAL IMAGE SEGMENTATION

Umut Ozertem[1], *Deniz Erdogmus*[1], *Tian Lan*[2]
[1]CSEE Department, Oregon Health & Science University, Portland, Oregon, USA
[2]BME Department, Oregon Health & Science University, Portland, Oregon, USA

ABSTRACT

Segmentation is a fundamental problem in image processing having a wide range of applications. Image segmentation algorithms in the literature range from a cost criterion based optimization techniques to various heuristic methods. In this paper, we propose utilizing mean shift spectral clustering for perceptually better image segmentation results.

1. INTRODUCTION

Image segmentation is a fundamental problem in image processing with a wide range of applications including feature extraction, filtering of noisy images, object recognition, and object-based video or image coding. The problem in image processing is defined as partitioning the image into distinct regions such that each region is homogenous and none of the unions of adjacent regions are homogenous.

Image segmentation techniques can be classified into four main groups. *Edge-based approaches* detect the edges in the images and link them to build contours; however, these methods are only applicable when the pixel intensity value itself is a suitable feature for segmentation [1]. *Split and merge approaches* partitions the image into primitive regions and use a similarity measure to merge neighboring regions until a predefined stopping criteria has been reached [2]. *Region-based approaches*, like region growing, have the advantage of low computational cost. On the other hand, results obtained by these methods are quite sensitive with respect to the chosen parameter values [3]. *Clustering-based approaches* are generally non-parametric or have few parameters. Usually, the problem of setting thresholds has been overcome by using a clustering based approach.

Introduced by Fukunaga and Hostetler [4], mean shift is a non-parametric clustering approach that seeks modes of a probability density function represented by a finite number of samples. Mean shift gained popularity after the formulation was revisited by Cheng [5], who applied the algorithm to clustering problem in an elegant way. Being an unsupervised learning algorithm, image segmentation is a natural application field for mean shift clustering. The shortcoming of mean shift is that the results are not always perceptually important. In mean shift, the number of clusters is automatically obtained for any given kernel function, and the segmentation results strictly depend on the choice of the kernel. The problem of removing perceptually unimportant clusters has been addressed before using heuristic methods that threshold the number of the pixels in each segment or check similarities between neighboring clusters using predefined thresholds. In this paper, we propose a principled way to segment images by measuring pdf distances between all pairs of mean shift results and remove/merge perceptually unimportant clusters to provide the final clustering.

2. THE PROPOSED METHOD

In this section, starting with the definition of the mean shift algorithm, the details of the proposed method will be discussed. Mean shift is a mode detection procedure based on probability density gradient of the data. For a given kernel function $K_\sigma(.,.)$, the kernel density estimate (KDE) becomes,

$$p(\mathbf{x}) = (1/N)\sum_{i=1}^{N} K_{\sigma_i}(\mathbf{x}-\mathbf{x}_i) \quad (1)$$

Using (1), the gradient of the probability density of the data is estimated and the local maxima points $\mathbf{y}_c$ are obtained. At these points, the gradient becomes null and the Hessian is negative (semi-)definite:

$$\nabla \hat{p}_K(\mathbf{y}_c) = 0 \qquad \nabla^2 \hat{p}_K(\mathbf{y}_c) \le 0 \quad (2)$$

The mean shift iterations are simply fixed-point iterations towards these stationary points. The volume that includes only the set of points that converge to the same mode after these fixed-point iterations is defined as the attraction basin and mean shift maps all the data samples to the local maxima of their corresponding attraction basin.

To be able to utilize this mean shift clustering based image segmentation approach, first the image should be mapped into a suitable feature space. A convenient selection for the features is the pixel coordinates and the intensity values for each color channel in the image. Other features like directional derivatives or any feature that can be defined in a pixelwise manner and can be utilized for segmentation. Regionwise defined features like shape or texture parameters can be added into the feature set by assigning those features to all pixels in the corresponding region. This mapping step is followed by density estimation, and finally, the resulting segmentation algorithm is based on a clustering in the selected feature domain. Feature selection

will naturally affect the results, and there may be alternative more informative feature definitions for specific areas of interest. The method can be applied to any feature set; however, for generality, in our computer simulations we used pixel coordinates and intensity values as features.

Motivated by the relationship between spectral clustering and density estimation [6] we propose using a two-step clustering algorithm for image segmentation. The first step determines the modes of the density estimate of the data with a fixed-point iterative procedure similar to mean shift, and the second step employs spectral clustering on a reduced size affinity matrix that defines similarities between the modes of the density. Typically, the number of modes M is much smaller as compared to the original data size N, and since the mean shift procedure is $O(N^2)$, the spectral clustering step is computationally negligible with $O(M^3)$.

2.1. Decision Boundary for Segmentation

In the Bayesian sense, optimal results for a classification problem can be obtained by minimizing the Bayes risk function for the given data. The probability of error is a widely accepted Bayes risk function, and the optimal decision boundary for the two-class case is given by $p_1q_1(\mathbf{x})=p_2q_2(\mathbf{x})$. In a clustering problem, however, the individual class/cluster densities are not available, and the overall data distribution is given by $p(\mathbf{x})=p_1q_1(\mathbf{x})+p_2q_2(\mathbf{x})$. The mean shift step inherently determines the boundaries between the attraction basins of all modes present in $\hat{p}_K$, the kernel-based estimate of this distribution. Presented in Figure 1 for a two-dimensional case, the local minimum of the overall distribution between the modes is a reasonable approximation to the Bayes boundary. For example, in the one-dimensional scenario, the separation boundary satisfies

$$\nabla\hat{p}_K(\mathbf{y}_c)=0 \qquad \nabla^2\hat{p}_K(\mathbf{y}_c)\geq 0 \tag{3}$$

Note that two modes, which are *supposed* to be in the same cluster can be partitioned artificially by this algorithm. Addressed by the following spectral clustering step, this shortcoming will be discussed in section 2.5.

2.2 Kernel Density Estimation

Since data distributions tend to take complex forms in many applications, determining a suitable parametric family to be used in a parametric method might become a tedious task. On the other hand, nonparametric methods based on sample spacing yield non-differentiable estimates. Producing continuously differentiable pdf estimates, kernel density estimation (KDE) provides an effective method for obtaining a density estimate that is suitable for gradient-based adaptive learning.

As with kernel-based methods, the selection of a suitable kernel function is central to the approach, and the literature on nonparametric KDE clearly indicates that the kernel function should be selected to match the distribution of the data as much as possible. At this point variable size

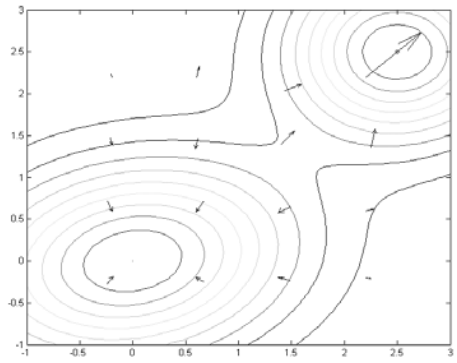

Figure 1. Contour plot for the overall probability density for two Gaussian clusters. Blue arrows represent the gradient field of the distribution.

kernel density estimation merits special attention due to its fast asymptotic behavior. Introducing individual kernel sizes for each data point will increase the overall computational load; on the other hand, it will also increase the performance by yielding a density estimate, which is less sensitive to outliers and more tuned to local scales in the data. The variable kernel size σ_i is selected such that it becomes larger for samples that don't have close neighbors; that is, it is more likely to be an outlier. Using the median of K nearest neighbor distance with a spherical Gaussian kernel or covariance of K nearest neighbor with an anisotropic Gaussian kernel are some possible choices for σ_i.

2.3 Mean Shift Iterations

The mean shift algorithm is used here to obtain an intermediate clustering result to be refined in the spectral analysis step, where the modes of the data distribution provide a natural intermediate clustering solution; hence, the data points in the same attraction basin form an intermediate cluster associated with the corresponding mode.

Starting with the kernel density estimate definition given in (1), one can derive the fixed-point iterations easily using the fact that at the peak of each mode the gradient of the density becomes zero, which yields

$$\frac{\partial p(\mathbf{x})}{\partial \mathbf{x}}^T = \frac{1}{N}\sum_{i=1}^{N}\frac{\partial K_\sigma(\mathbf{x}-\mathbf{x}_i)}{\partial \mathbf{x}} = \mathbf{0} \tag{4}$$

Reorganizing the terms in (7) and solving for $\mathbf{x}$ specifically, for a Gaussian kernel, one can obtain the update equation as

$$\mathbf{x} \leftarrow \left(\sum_{i=1}^{N} G_{\sigma_i}(\mathbf{x}-\mathbf{x}_i)\mathbf{x}_i\right) \Big/ \left(\sum_{i=1}^{N} G_{\sigma_i}(\mathbf{x}-\mathbf{x}_i)\right) \tag{5}$$

The computational load of this step is $O(N)$ per sample per iteration. In practice, all samples require a different number of iterations to converge and to be able to reduce the computational load, a stopping criterion can be checked for each sample individually, which leads to a decrease in the number of the samples that need to be updated. Employing finite support kernels instead of Gaussian kernels or utilizing Fast Gauss Transform [7] to

approximate the iteration update are other approaches to reduce the computational cost. Due to the existence of statistical variance in the density estimation resulting from the finite sample effects and the possibility of existence of multi-modal clusters, in general, results of mean shift needs to be refined. The proposed steps will be discussed in the following subsections.

2.4. The Normalized Mode Affinity Matrix

Associating each data sample with a mode of the density estimate using mean shift, an affinity matrix needs to be define to summarize the pair-wise affinities between each mode. According to the connection between kernel affinity measure based methods and KDE [6], the affinity matrix entries are given by the convolution of the kernels associated with the given samples. In the case of variable size Gaussian KDE, for kernel sizes σ_i and σ_j, this results in the following affinity between these samples for the mode affinity measure

$$\mathbf{G}_{ij} = G_{\sqrt{\sigma_i^2+\sigma_j^2}}(\mathbf{x}_j - \mathbf{x}_i) \tag{6}$$

A number of distance measures can be employed to define the mode affinity matrix, including Euclidean distance between the distributions of different modes or information theoretic divergence measures. Employing the Euclidean distance as the correlation measure, the affinity between mode *i* and *j* is given by

$$D_{ij} = \int p_i(\mathbf{x}) p_j(\mathbf{x}) d\mathbf{x} \tag{7}$$

where $p_i(\mathbf{x})$ and $p_j(\mathbf{x})$ are the density functions of the corresponding modes. Substituting the KDE definition given in (1) one can rewrite the affinity measure between the mode pair as

$$\hat{D}_{ij} = \frac{1}{N_i N_j} \sum_k \sum_l G_{\sqrt{\sigma_k^2+\sigma_l^2}}(\mathbf{x}_k^i - \mathbf{x}_l^j) \tag{8}$$

where $\mathbf{x}_k^i$ denotes the k^{th} sample associated with mode *i*. Recalling the characteristics of graph cut and normalized graph cut, a more numerically stable affinity measure between modes *i* and *j* can be defined as follows

$$\tilde{\mathbf{G}}_{ij} = \frac{\hat{D}_{ij}}{\sqrt{\hat{D}_{ii}}\sqrt{\hat{D}_{jj}}} \tag{9}$$

At this point, one should also notice that in the function space defined according to the Euclidean inner product definition, and the measure defined in (9) is the *angle* between the two distributions $p_i(\mathbf{x})$ and $p_j(\mathbf{x})$.

2.5. Connected Components of the Mode Affinity Matrix

Any standard spectral clustering method in the literature can be applied to the mode affinity matrix to obtain the final clustering results, and it is important to note that the computational cost required to determine eigenvectors of this matrix is $O(M^2)$ per eigenvector, which is negligible as compared to the $O(N)$ per sample per iteration computational load mean shift. As well as applying different spectral clustering methods from the literature, we also propose utilizing a robust algorithm based on connected components analysis. Having a $O(M^4)$ complexity, this algorithm is impractical for the dataset itself; however, the approach is simple and produced good results for the small-sized mode affinity matrices.

The procedure of determining the connected components can be summarized as follows. First, all the affinities in the mode affinity matrix $\tilde{\mathbf{G}}$ are sorted in descending order. Next, the weakest connection is removed and the graph connectivity is checked. This procedure is iterated until a predefined number of components in the graph is reached. Performed in each iteration with $O(M^2)$ complexity, checking the graph connectivity is the dominant computational load of this approach, resulting in a $O(M^4)$ complexity for the overall algorithm. To check the graph connectivity, a well-known connected components algorithm is used [8].

Instead of defining the number of clusters as a preset value, one can also define similar methods by defining a threshold for the affinity values between pairs, which will automatically determine the number of clusters. In this method, selecting a suitable threshold can be achieved by observing the clustering structure while increasing this threshold from 0 to 1. The clusters that remain unchanged for a larger interval in this experiment can be regarded as statistically significant or natural. This procedure has been previously employed for setting temperature and kernel size in clustering algorithms [9].

Constructing $\tilde{\mathbf{G}}$, one can have an idea of the distances between the modes of the overall distribution and the mode-affinity analysis step provides a systematic way of merging the modes by statistically investigating the possibility that neighboring modes might belong to the same cluster. Although it might be possible to change the kernel size to estimate clusters with a single mode for each one; however, the mode affinity analysis step defines a principled way of eliminating this requirement. In general, by evaluating the results provided by mean shift, the proposed algorithm provides a systematic approach for estimating perceptually and statistically important segments in the image.

3. EXPERIMENTAL RESULTS

In this section we will present simulation results and compare the results with the ones obtained by mean shift. The widely used baseball player image has been used here to enable further comparisons among the results of the some other algorithms in the literature. The features used in the experiments are the pixel coordinates and the intensity values; rephrasing what was mentioned before, one may choose more suitable features for specific cases; however, the aim of the paper is to demonstrate the contribution of the proposed algorithm, rather than optimizing results for a specific segmentation application. An important point for implementing a good density estimator is to normalize the

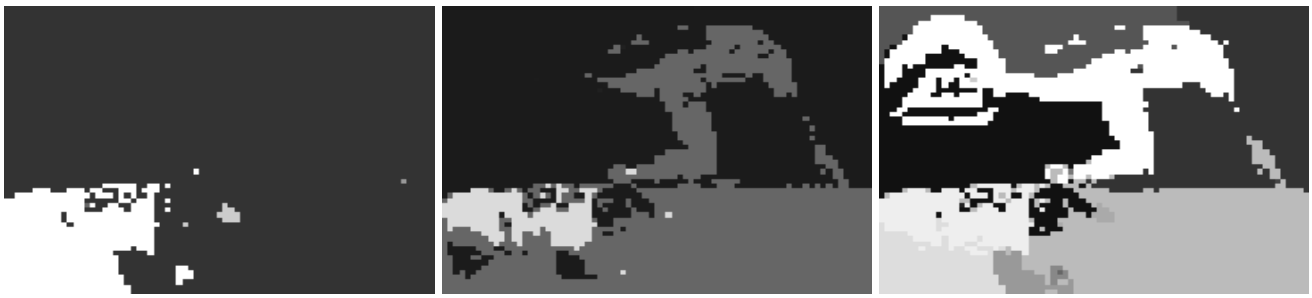

Figure 2.a Figure 2.b Figure 2.c

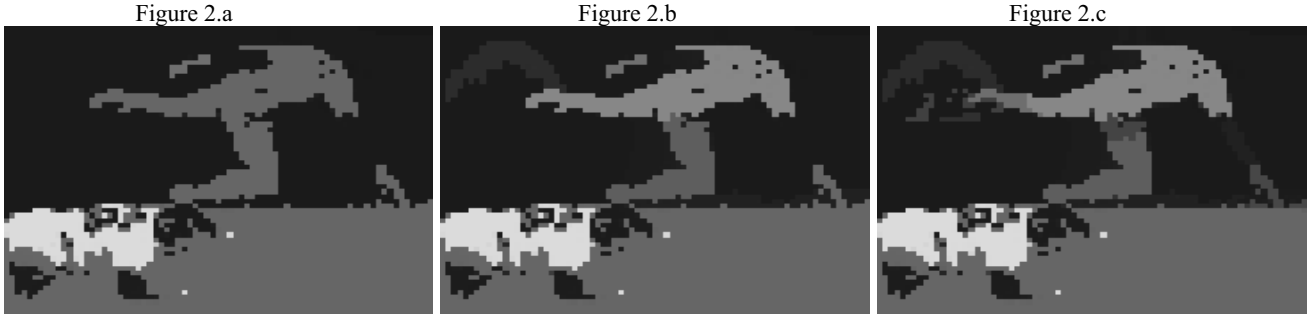

Figure 2.d Figure 2.e Figure 2.f

Figure 2. Results of the Normalized Cut algorithm for 5, 10,and 15 clusters are presented in (a), (b) and (c). Results of the Mean Shift Spectral Clustering are presented in (d), (e), and (f) for the same number of clusters, respectively.

data along each feature axis globally to be able to effectively use the spherically symmetric kernels by choosing the kernel size as median of the neighbor distances. This requirement is relaxed in the case of utilizing neighbor covariance instead of distance for anisotropic kernels, and note that, this leads to a more efficient estimate as compared to the isotropic kernels, even if the dataset is normalized.

Figure 2 depicts segmentation results for different number of clusters. Particularly for this image, for 15 and more output clusters both algorithms perform similarly; however, for smaller number of output clusters, unlike mean shift spectral clustering results, results obtained by normalized cuts deviates from perceptually meaningful clusters. The original image is shown in Figure 3, downsampled and converted to grayscale.

4. CONCLUSIONS

In this paper mean shift spectral clustering, namely a mean shift algorithm with a spectral clustering based post processing, is utilized for perceptually better results in image segmentation. A variable size kernel density estimate has been employed for this purpose due to its well-known characteristics of fast asymptotic behavior and yielding a density estimate that is less dependent to the outliers in the data. With a negligible amount of additional computation, spectral analysis step provided perceptually more important clusters as compared to the traditional mean shift algorithm.

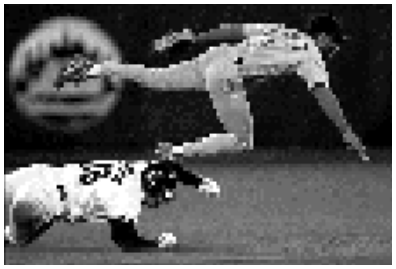

Figure 3. The baseball players segmentation benchmark image.

5.ACKNOWLEDGEMENTS

This work was supported by DARPA under contract DAAD-16-03-C-0054 and by NSF under grant ECS-0524835.

6. REFERENCES

[1] J.F. Canny, “A Computational Approach to Edge Detection,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 8, pp.679-698, 1986.

[2] R. Pal, S.K. Pal, “A Review in Image Segmentation Techniques,” Pattern Recognition, vol. 26, pp.1277-1294, 1993.

[3] K.S. Fu, J.K. Mei, “A Survey on Image Segmentation,” Pattern Recognition, vol. 13, pp. 3-16,1981.

[4] K. Fukunaga, L.D. Hostetler, “The Estimation of the Gradient of a Density Function, with Applications in Pattern Recognition,” IEEE Transactions on Information Theory, vol. 21, pp. 3240, 1975.

[5] Y. Cheng, “Mean Shift, Mode Seeking, and Clustering,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 17, no. 8, pp. 790-799, 1995.

[6] R. Jenssen, D. Erdogmus, J. C. Principe, T. Eltoft, “The Laplacian PDF Distance: A Cost Function for Clustering in a Kernel Feature Space,” Advances in Neural Information Processing Systems, pp.625-632, 2004.

[7] L. Greengard, J. Strain, “The Fast Gauss Transform,” SIAM Journal of Scientific and Statistical Computing, vol. 12, no. 1, pp. 79–94, 1991.

[8] T.H. Cormen, C.E. Leiserson, R.L. Rivest, *Introduction to Algorithms*, MIT Press and McGraw-Hill, New York, 1990.

[9] M. Blatt, S. Wiseman, E. Domany, “Data Clustering Using a Model Granular Magnet,” Neural Computation, vol. 9, no. 8, pp. 1805-1842, 1997.