IMPROVING SUPERPIXEL-BASED IMAGE SEGMENTATION BY INCORPORATING COLOR COVARIANCE MATRIX MANIFOLDS

Xianbin Gu, Jeremiah D. Deng, Martin K. Purvis

Department of Information Science, University of Otago, New Zealand

ABSTRACT

We propose to use color covariance matrices of superpixels as a feature in addition to colors. A non-Euclidean distance metric is employed for the covariance matrix manifolds. We then introduce three ways of fusing the similarity matrices obtained from both feature spaces for affinity graph generation. Experiments carried out using a benchmark dataset reveals that our approach achieves competitive and even better results compared with the state of the art.

Index Terms— image segmentation, superpixel, covariance matrices, bipartite graph

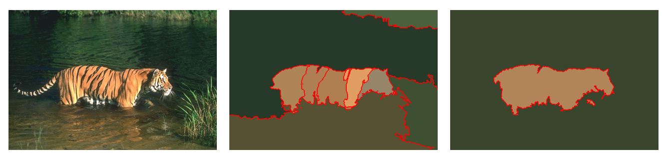
1. INTRODUCTION

Superpixels, i.e., clusters of image pixels grouped together based on their perceptual similarity, are finding more and more applications in image processing. A number of clustering algorithms can be used to generate superpixels, e.g., density-based algorithms such as Mean Shift [1], and graph-based methods such as FH [2] and Power watersheds [3]. Meanwhile, state-of-the-art implementations such as SLIC [4] can form image superpixels very effectively as they are specially designed for the purpose. Region-based features can be extracted from superpixels, which have found applications in object classification and localization [5]. For image segmentation, since superpixels are usually over-segmented, some further treatment is necessary for larger but homogeneous regions to be formed as the segmentation outcome [6]. Notably, there is a growing trend in treating superpixels as cues to be merged through spectral clustering [7][8]. More recently, Wang et al. [9] applied a sparse-coding method to superpixels in a ℓ_0 space, and by using a modified affinity matrix of the Transfer Cut algorithm [7], achieved some impressive results.

In a wider context, it is found that the fusion of multiple cues can lead to better segmentation, e.g., combining color histograms, local binary patterns feature, and Bag of Words [10]. Apparently a suitable representation of superpixels may improve the quality of superpixel-based image segmentation.

Following these former studies, in this paper we look at improving superpixel feature representation by employing color covariance matrices in addition to superpixel color features. Unlike conventional color or texture features, the covariance matrices are not embedded in an Euclidean space but define a manifold, so a Riemannian metric is adopted. Using the same graph-cut approach as in [7], we then propose several ways of fusing similarity measured in two difference spaces, i.e., color and color covariance, when constructing the affinity graph. These methods are then tested in the experiments, and the final segmentation outcome is compared with the state-of-the-art algorithms.

Fig.1 gives a quick comparison of our algorithm and SAS [7]. Note that the tiger is split into different chunks by SAS, but not by our method.



(a) (b) (c)

Fig. 1. Visual comparison of the best segmentation: (a) original image; (b) SAS; (c) our method.

Next, Section 2 describes the feature extraction process, followed by the construction of the affinity graph in Section 3. Section 4 reports the experiment results. Finally, we conclude the paper.

2. FEATURE EXTRACTION

Superpixels can be generated by using Mean Shift or FH algorithms under different parametric settings. Our first focus is on extracting useful features from the generated superpixels based on which image segmentation can be better carried out.

Following previous work, first we use the average color as the feature for each superpixel. In particular, the CIE Lab color space is adopted because of its good approximation to human vision. Denote a pixel vector in the Lab space as $\mathbf{x}$. Each superpixel, denoted as set $S_i = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_M\}$ (M is the number of pixels included in S_i), is then represented by its average color vector

$$\mathbf{c}_i = E(\mathbf{x}_m), \mathbf{x}_m \in S_i \quad (1)$$

The superpixels of an image are then regarded as a set of points in a 3-D Lab space with an Euclidean distance metric.

Empirically, using color information alone may not be enough for generating perfect image segmentation. Other non-Euclidean feature spaces could also be considered. In particular, we are interested in using the color covariance within superpixels, defined as

$$\Sigma_i = E((\mathbf{x}_m - \mathbf{c}_i)(\mathbf{x}_m - \mathbf{c}_i)^T), \mathbf{x}_m \in S_i \tag{2}$$

Covariance matrices as a tensor lie on a smooth manifold, hence requiring a non-Euclidean distance metric. Since they are symmetric and positive semi-definite, we can use the Förstner & Moonen metric [11] given as

$$d(\Sigma_A, \Sigma_B) = \sqrt{\sum_{r=1}^{n} \ln^2 \lambda_r}, \tag{3}$$

where Σ_A, Σ_B are two covariance matrices of dimension $n \times n$, and $\lambda_r(r = 1, \cdots, n)$ are eigenvalues from the generalized eigenvalue problem $|\lambda\Sigma_A - \Sigma_B| = 0$. In our case, the covariance matrices are of 3×3 dimension, so the cost of eigenanalysis cost is negligible. Assume the average size of superpixels is k pixels, the overall computational complexity so far introduced for each superpixel hence is $O(k)$.

3. GRAPH-BASED SEGMENTATION

3.1. Segmentation with superpixel-based graph cut

We use Transfer Cuts (Tcut) [7], a spectral graph-based partition algorithm, to perform the image segmentation. First a bipartite graph $G(X, Y, B)$ is constructed over the pixels and superpixel sets of the input image I. Here X denotes the vertices with value of pixels and superpixels, Y denotes the vertices representing superpixel sets created by different or the same superpixel algorithms with different parameter settings, and $B = [W_X, W_Y]^T$ holds the edges of the bipartite graph. Then, the cluster structure of the bipartite graph G can be found by singular vector decomposition (SVD) of the normalized across-affinity matrix $\widehat{B} = D_X^{-\frac{1}{2}} B D_Y^{\frac{1}{2}}$, where $D_X = \text{diag}(B\mathbf{1})$, $D_Y = \text{diag}(B^T\mathbf{1})$.The left and right singular vectors of $\widehat{B}$ contain the partition information of vertices in X and Y respectively.

The Tcut algorithm finds the partition of X more efficiently by solving an equivalent eigen-problem [7]:

$$L_Y \mathbf{v} = \lambda D_Y \mathbf{v}, \tag{4}$$

where $L_Y = D_Y - W_Y$, $W_Y = \widehat{B}^T D_X^{-1} B$, and $\mathbf{v}$ is the eigenvector.

3.2. Construction of the affinity graph

One major difference between our approach and other related work is in the construction of the affinity graph . In [7], each superpixel is connected with the nearest neighborhood among its spatially adjacent superpixels, which fails to catch the relationship of those vertices that are separated by spacial distance but close in the feature space. In [9], this weakness was overcome by measuring the similarity of the superpixels with their ℓ_0 sparse-coding representation. However, this problem can be solved in another way. Our approach is to seek additional information that better represents the superpixels, so that the spatial constrains can be removed. This is achieved by using the covariance matrices as the complementary feature to color features.

Let d_{ij} denotes the distance between superpixel S_i and S_j. Depending on the features being used, d_{ij} can be the Euclidean distance in the Lab space, or the non-Euclidean metric in (3). The similarity w_{ij} between the two superpixels then is defined as follows:

$$w_{ij} = \begin{cases} e^{-\beta \min{(d_{ij}, d_{ji})}} & \text{if } i \neq j \\ 1, & \text{otherwise} \end{cases} \tag{5}$$

where β is a coefficient of the Gaussian-like kernel, and d_{ij} is normalized into [0,1]. Note that d_{ij} may be unequal to d_{ji} because of the normalization.

Now we have two different similarity matrices, W^c and W^Σ, representing the two different feature spaces of the superpixels, i.e., color and color covariance respectively. To fuse these two similarity matrices so as to construct an affinity graph, one solution is by means of the entry-wise product (*aka* Hadamard product):

$$W_{\text{HP}} = W^c \circ W^\Sigma. \tag{6}$$

There are alternatives that can be adopted. de Sa [12] used direct matrix product to similarity matrices from two views:

$$W_{\text{DP}} = W^c W^\Sigma, \tag{7}$$

In addition, Joachims [13] combined two individual modalities by simply adding them together:

$$W_{\text{AD}} = W^c + W^\Sigma, \tag{8}$$

Finally, the affinity graph is built by connecting the pixels to their superpixels and pairing the superpixels that are most similar to each other (within each over-segmented image), i.e., building W_X with the weight on each edge is set to a constant, and forming W_Y with k-NN. Our algorithm is summarized in Table 1.

4. EXPERIMENTS

For the purpose of comparison, the evaluation of our algorithm is done on a standard benchmark image segmentation

Table 1. Algorithm

Steps		
1	Superpixel generation	Use Mean Shift and FH algorithms with different parameters to create over segmentation of the image;
2	Feature extraction	Compute the average color value of the superpixels and their covariance matrix by (2);
3	Similarity measurement	Use (5) to compute the similarity of superpixels, and fuse W^c and W^Σ by (6), (7), or (8);
4	Graph construction	Define a bipartite graph $G(X, Y, B)$ over pixels and superpixels, and build the similarity matrix $B = [W_X, W_Y]^T$;
5	Segmentation	Partition graph G with the Transfer Cut algorithm.

dataset, the Berkley Segmentation Dataset (BSD) [14], and the parameters are set the same as in [7] and [9]. Specifically, the superpixels are created by Mean Shift and FH algorithms; The weights on edges of W_X are fixed to 1×10^{-3}, and set $\beta = 20$ for the distance-tuned Gaussian kernel in W_Y. We use a nearest-neighbor graph for W_Y, with $k = 1$.

The evaluation is measured by four indicators: Probabilistic Rand Index (PRI) [15], Variation of Information (VoI) [16], Global Consistency Error (GCE) [17], and Boundary Displacement Error (BDE) [18]. For PRI, a higher value means better result; for the rest the lower the better. The average scores of the four indicators are reported for comparison.

The experiments are conducted in two parts. First, the same as in [7], we manually set the number of segmentations of every image to find the best performance of the algorithms for comparison. Secondly, following [9], we fix the segment number $K = 2$, which is more practical in real applications. In both scenarios our method gives competitive performance. We examine the performance of all three fusion methods in our experiments, and the results are listed in Table 2 and Table 3. Our method ranks the first place with PRI and VoI when the cluster number K is manually set, and when K is fixed to 2, it gets the best scores in PRI, VoI, and GCE. The scores of the SAS algorithm and ℓ_0-sparse representation methods are obtained from [7] and [9].

As shown in Table 3, the choice of the color space does not seem to be critical, since the utilization of the color covariance matrices seems to boost the performance significantly to a competitive level even for RGB and HSV. SAS, on the other hand, reports worse results in VoI and GCE when using these two color spaces compared with using Lab.

The experiment results on the BSD dataset show that the new superpixel feature extracted by a covariance matrix apparently improves the average performance of the bipartite graph based algorithm when combined with color cues. By removing the spatial constraints, our method seems to handle long-range homogeneity well, forming superpixels well aligned with object contours. The Hadamard product seems to performs the best among the three fusing schemes, but the difference is mostly marginal.

Table 2. Performance comparison over the BSD database with K adjusted manually

Algorithms	PRI	VoI	GCE	BDE
SAS [7]	0.8319	1.6849	**0.1779**	11.2900
ℓ_0-sparse-coding [9]	0.8355	1.9935	0.2297	**11.1955**
Ours (using W_{HP})	**0.8495**	**1.6260**	0.1785	12.3034
Ours (using W_{DP})	0.8345	2.1169	0.2341	12.0008
Ours (using W_{AD})	0.8397	2.0359	0.2308	11.8868

Table 3. Performance comparison over the BSD database with K fixed to 2

Algorithms	PRI	VoI	GCE	BDE
SAS [7]	0.6179	2.011	0.1106	42.2877
ℓ_0-sparse-coding [9]	0.6270	2.0299	0.1050	**23.1298**
Ours (using W_{HP})	**0.6312**	**1.9350**	**0.0820**	35.8760
Ours (using W_{DP})	0.5998	2.0336	0.0892	29.1803
Ours (using W_{AD})	0.6284	1.997	0.0940	24.6991
RBG (SAS)	0.6189	2.0224	0.1138	42.5141
HSV (SAS)	0.6182	2.0450	0.1203	42.0903
RBG (using W_{HP})	0.6289	1.9426	**0.0815**	33.9353
HSV (using W_{HP})	**0.6317**	1.9549	0.0838	30.2480

Fig.2 shows more experimental results of our algorithm when K is set to 2. The method seems to be quite effective in foreground-background separation.

5. CONCLUSION

We present a graph-based segmentation approach that utilizes color covariance matrices to boost the performance of graph-based image segmentation. A non-Euclidean metric is employed for the covariance matrix space, and the new feature is then integrated with color information to form the affinity graph for segmentation. Our empirical results show that the new approach produces better or competitive segmentation results compared with the state-of-the-art approaches. It is not sensitive to the choice of color space, different from previous work. In the future, we intend to incorporate some additional features with our algorithm.

6. REFERENCES

[1] Dorin Comaniciu and Peter Meer, "Mean shift: A robust approach toward feature space analysis," *TPAMI*, vol. 24, no. 5, pp. 603–619, 2002.

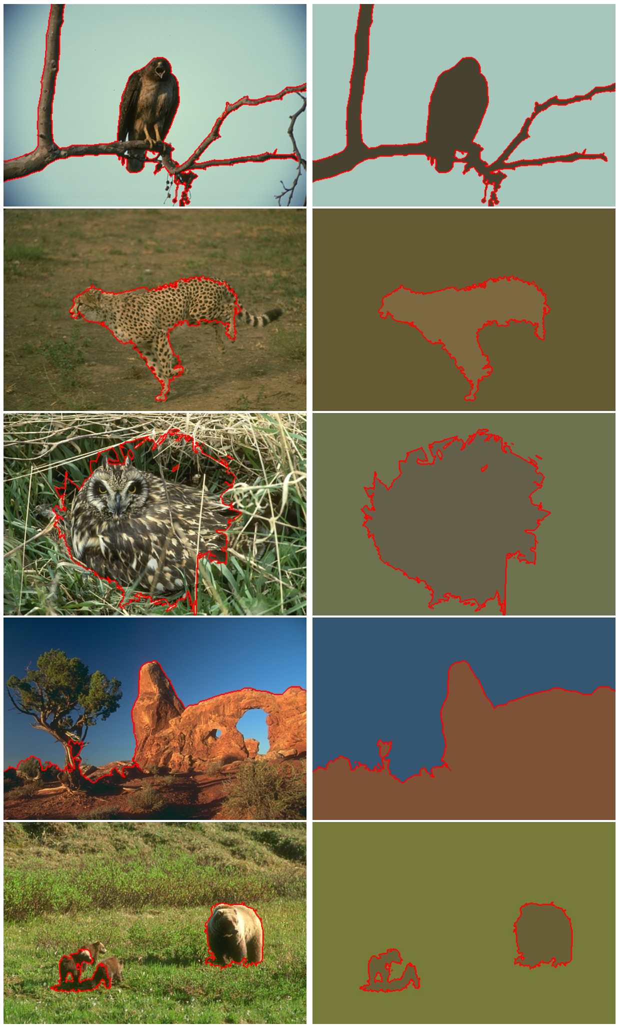


Fig. 2. Some more results of our method ($K = 2$).

[2] Pedro F Felzenszwalb and Daniel P Huttenlocher, "Efficient graph-based image segmentation," *IJCV*, vol. 59, no. 2, pp. 167–181, 2004.

[3] C. Couprie, L. Grady, L. Najman, and H. Talbot, "Power watersheds: A new image segmentation framework extending graph cuts, random walker and optimal spanning forest," in *ICCV'09*, 2009, pp. 731–738.

[4] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk, "Slic superpixels compared to state-of-the-art superpixel methods," *TPAMI*, vol. 34, no. 11, pp. 2274–2282, 2012.

[5] Brian Fulkerson, Andrea Vedaldi, and Stefano Soatto, "Class segmentation and object localization with superpixel neighborhoods," in *ICCV'09*, 2009, pp. 670–677.

[6] C. Panagiotakis, H. Papadakis, E. Grinias, N. Komodakis, P. Fragopoulou, and G. Tziritas, "Interactive image segmentation based on synthetic graph coordinates," *Pattern Recognition*, vol. 46, no. 11, pp. 2940–2952, 2013.

[7] Zhenguo Li, Xiao-Ming Wu, and Shih-Fu Chang, "Segmentation using superpixels: A bipartite graph partitioning approach," in *CVPR'12*, 2012, pp. 789–796.

[8] Tae Hoon Kim, Kyoung Mu Lee, and Sang Uk Lee, "Learning full pairwise affinities for spectral segmentation," in *CVPR'10*, 2010, pp. 2101–2108.

[9] Xiaofang Wang, Huibin Li, Charles-Edmond Bichot, Simon Masnou, and Liming Chen, "A graph-cut approach to image segmentation using an affinity graph based on ℓ_0-sparse representation of features," in *ICIP'13*, 2013.

[10] Bin Cheng, Guangcan Liu, Jingdong Wang, Zhongyang Huang, and Shuicheng Yan, "Multi-task low-rank affinity pursuit for image segmentation," in *ICCV'10*, 2010, pp. 2439–2446.

[11] Wolfgang Förstner and Boudewijn Moonen, "A metric for covariance matrices," in *Geodesy - The Challenge of the 3rd Millennium*, ErikW. Grafarend, FriedrichW. Krumm, and VolkerS. Schwarze, Eds., pp. 299–309. Springer, 2003.

[12] Virginia R de Sa, "Spectral clustering with two views," in *ICML workshop on learning with multiple views*, 2005.

[13] Thorsten Joachims, "Transductive learning via spectral graph partitioning," in *ICML'03*, 2003, pp. 290–297.

[14] Pablo Arbelaez, Michael Maire, Charless Fowlkes, and Jitendra Malik, "Contour detection and hierarchical image segmentation," *TPAMI*, vol. 33, no. 5, pp. 898–916, 2011.

[15] Ranjith Unnikrishnan, Caroline Pantofaru, and Martial Hebert, "Toward objective evaluation of image segmentation algorithms," *TPAMI*, vol. 29, no. 6, pp. 929–944, 2007.

[16] Marina Meilă, "Comparing clusterings: an axiomatic view," in *ICML'05*. ACM, 2005, pp. 577–584.

[17] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *ICCV'01*, 2001, vol. 2, pp. 416–423.

[18] Jordi Freixenet, Xavier Muñoz, David Raba, Joan Martí, and Xavier Cufí, "Yet another survey on image segmentation: Region and boundary information integration," in *ECCV'02*, pp. 408–422. Springer, 2002.