UNSUPERVISED TEXTURE SEGMENTATION USING FEATURE SELECTION AND FUSION

Suranjana Samanta and Sukhendu Das

Department of Computer Science and Engineering,
Indian Institute of Technology - Madras, Chennai - 600036, India.
{ssamanta@cs., sdas@}iitm.ac.in

ABSTRACT

This paper describes a method of unsupervised color texture segmentation by efficiently combining different features obtained from multi-channel and multi-resolution filters. The DWT and DCT features are extracted separately from 3 color bands of the image and then fused together for optimal performance. The features are then ranked according to a selection criteria. We propose a new correlation measure for the task of feature ranking. To select the best combination of features to be used, we use the property of cluster scatter of a selected set of features. Finally, the optimum number of ranked order features are used for segmentation using a Fuzzy C-Means classifier. The performance of the proposed segmentation method is verified using standard benchmark datasets.

Index Terms— feature fusion, selection, ranking, FCM, correlation.

1. INTRODUCTION

Texture plays an important role in low-level image analysis and understanding. Classification and segmentation of texture content in digital images has received considerable attention during the past two decades and numerous approaches have been presented [1], [2], [3], [4], [5] and [6]. The focus of this paper is to implement an unsupervised method of feature selection and fusion for color texture segmentation to improve the performance. Commonly used texture features are based on Gabor filter bank [2], [5], GLCM based features [5], DWT and DCT [2]. A comparative study of various filters for texture classification has been presented in [1]. One important result is that the wavelet features perform better than that extracted using Gabor filter bank. A combination of features extracted by Gabor filter bank and DWT produce better segmented results on texture images [2].

However, combining features extracted by different mechanisms does not always improve accuracy. Increasing the number of features without proper selection leads to the problem of "curse of dimensionality". Some features with overlapping class-wise distribution confuse the classifier leading to degraded accuracy. Features which highly resemble (say, transformed) some other selected features are redundant and should be avoided, as they do not contribute any new information. These types of features must be identified and removed. A lot of work has been done on feature selection. Liu and Yu [4] provide a survey of different feature selection techniques, which are divided into four categories. The challenge is to choose the appropriate feature selection criteria for a given domain and obtain the optimum number of features for improved classification accuracy. Clausi and Deng [7] have used PCA as a feature reduction technique in case of texture segmentation. Though PCA is very popular, it is mainly a dimensionality reduction method rather than a feature selection technique. In case of supervised feature selection technique, Cohen-Kappa [3] and classification accuracy are used as feature selection criteria. Mitra, Murthy and Pal [8] have used Maximal Information Compression Index for unsupervised feature selection, which has been used in [6] for texture segmentation. However, less study has been done on selecting the optimum number of features to provide better segmentation accuracy, than using the whole set of features. Selection criteria like KL Divergence, entropy, mutual information, correlation coefficient and other cluster compactness properties have been used as feature selection criteria [4], [9], mostly in cases of supervised classification.

The overall methodology for unsupervised texture segmentation is shown in Fig. 1. The texture features are first

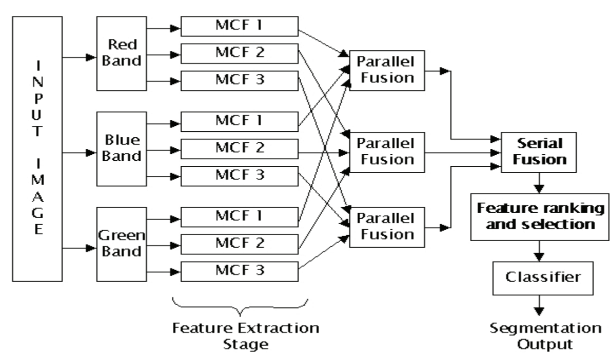

Fig. 1. **Stages of processing for Texture Segmentation**

extracted from three different bands of an RGB texture image. The features are then fused together using parallel and serial fusion techniques [5]. The fused features are then ranked according to a selection criteria. Next, we find the optimum number of features to be selected and fed to an unsupervised classifier for segmentation. Results are shown using benchmarked datasets and a few texture images.

2. TEXTURE FEATURE FUSION

This section describes the process of feature fusion for a relevant set of texture features used for segmentation. Multichannel filters (MCF) have been used for texture feature extraction. Initially features are extracted from each of the three bands in a RGB color image. 8-tap Daubechies, DCT and a combination of Haar filters followed by DCT filtering have been used for feature extraction. Features obtained from the three color bands using each of these techniques, are fused together using an imaginary number representation. We take the magnitude of the complex features to obtain a new set of features. This technique of feature fusion is called Parallel Feature Fusion [5]. These parallely fused features are then concatenated together using serial feature fusion [5]. In the following, we describe the process of ranking the fused feature set, using our proposed measure of feature correlation.

2.1. FEATURE RANKING

Feature ranking is closely related to feature selection technique. The selection method deals with mainly two issues: feature selection criteria (a function) and a searching algorithm, which is used to find a subset of features to be evaluated. For a computationally efficient design, we use a simple and fast feature selection criteria, and a Sequential Forward Search (SFS) technique [9]. Since the class information is missing for unsupervised feature selection, we cannot determine if a highly uncorrelated feature is noisy or a valid feature distribution. A feature, which is highly correlated with other features most likely contains least amount of noise and hence selected as the first ranked feature. Let, F_{us} be the set of unselected features and F_s be the set of (rank-wise) selected features at some stage of iteration. At first stage of iteration ($F_s = NULL$), the first ranked feature is selected as

$$F_1 = \arg\max_i(V_i)\,, \;\; i \in F_{us} \tag{1}$$

$$\text{where,} \;\; V_i = \operatorname*{mean}_j[corr(f_i, f_j)|_{i \neq j}]\,, \;\; j \in F_{us} \tag{2}$$

$$\text{where,} \;\; corr(f_i, f_j) = \frac{cov(f_i, f_j)}{var(f_i) * var(f_j)} \tag{3}$$

$var(f_i)$ is the variance of the distribution produced by feature f_i and $cov(f_i, f_j)$ is the covariance between a pair of feature distributions f_i and f_j.For obtaining the feature for the k^{th} rank ($k \geq 2$), we select a feature from F_{us} as,

$$F_k = \arg\min_i[R_i]\,, \;\; i \in F_{us} \tag{4}$$

$$\text{where,} \;\; R_i = \operatorname*{mean}_j(F_{corr}(i,j))|_{i \neq j}\,, \;\; j \in F_s \tag{5}$$

$$\text{where,} \;\; F_{corr}(i,j)|_{(i \neq j)} = \frac{corr(f_i, f_j)}{\left| \frac{1}{var(f_i)} - \frac{1}{var(f_j)} \right|} \tag{6}$$

The i^{th} feature (in F_{us}) which minimizes the mean of the values of the function $F_{corr}(i,j)$ computed for all j, is considered as the next selected feature in the ranked set (F_s). This ensures that the selected feature is minimally redundant to the pre-selected set of features. We have evaluated this selection method using both simulated and real world databases. For simulated dataset, we have used 3 partially overlapping Gaussian distributions with varying standard deviations (chosen suitably) for overlap, along with a redundant feature and a random noise as the other 2 dimensions. Experiments over many trials revealed that the redundant feature is ranked 4^{th}, while the random noise is ranked 5^{th}. This ensures that the algorithm works correctly.

In case of real world dataset, we have taken datasets from the UCI repository [10]. We compared the proposed selection criteria with correlation and Maximal Information Compression Index [8]. Classification accuracy has been considered as the performance evaluation of these three techniques, which has been averaged over thirty random observations. The plot shown in Fig. 2 shows that the proposed method provides better classification accuracy than both the techniques, for Wine Dataset [10].

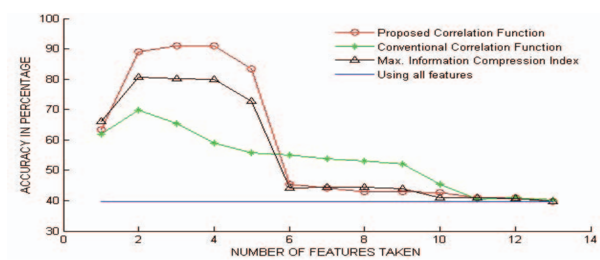

Fig. 2. **Comparison of proposed selection criteria with correlation and maximal information compression index.**

3. FEATURE SELECTION AND SEGMENTATION

To select the optimum number of features from the ranked feature set, we use the property of cluster scatter. In the absence of class labels (unsupervised segmentation), we assume that the spread (or variance) of the samples should neither be too compact nor wide for a useful feature distribution along a particular dimension. The former (compactness) may be caused due to large overlapping class-wise distributions and the later (very wide scatter) may be due to noise. The goodness of a k^{th} ranked feature being examined is estimated by calculating the mean of the correlation (Eqn. 3) values for all possible (k-1) feature pairs, which are formed by pairing the k^{th} ranked feature with the selected (k-1) list of features. To select the optimal number of rank-ordered features for the task of segmentation, we observe the mean of cluster correlation values with increasing number of features (k) in the cluster. At each stage of iteration, we consider a cluster space formed by the selected 'k-1' features in F_s and find the goodness measure for the resultant cluster formed by fusing the k^{th} feature; where k varies from 2 to 76. Any peak (or dip) in the goodness function identifies that the k^{th} feature is highly correlated (or uncorrelated) to some of the already selected set of features. We also assume that only a few of the low-ranked features, which confuse the classifier (low inter-class separability or high degree of overlap) and degrades the segmentation accuracy, must be rejected.

Fig. 3 shows a typical correlation plot, as an example to select the optimum number of features. The vertical arrow points to the heel of the curve, where there is a sharp change in gradient of the curve. The threshold to detect the sharp change from the gradient of the curve is obtained empirically. The features, which are ranked before the heel point are selected for the final segmentation. The features, which are ranked after the heel point are redundant or confusing (responsible for the globally sharp change in gradient of the curve) and are hence rejected. From this curve, we infer that the first 73 ranked features should be selected. Segmentation is done using Fuzzy C-Means (FCM) unsupervised classifier. Fig. 4(b)-(d) shows the segmented output using only one of the features. Fig. 4(e) shows the segmented output using all the features after fusion. Fig. 4(f) shows the segmented output using final selected feature set (first 73 ranked features). Table 1 shows the segmentation accuracy using different features, averaged over ten different texture images.

Table 1. **Accuracy (in %-age) of Segmentation for different feature sets, averaged over 10 images; D - dimension.**

Feature Set	D	Accuracy
Daubechies filter features	4	87.47
DCT filter features	36	87.91
Haar+DCT filter features	36	73.16
All features (without fusion)	228	85.24
After parallel & serial fusion	76	69.77
After feature ranking & selection	[61-73]	93.18

4. EXPERIMENTAL RESULTS

The proposed approach of unsupervised texture segmentation was tested on real world feature databases obtained from the UCI Repository [10] and color texture images. Fig. 5 shows the plots of accuracy in classification for Heart Disease and Lymphography datasets obtained from [10]. The curves show

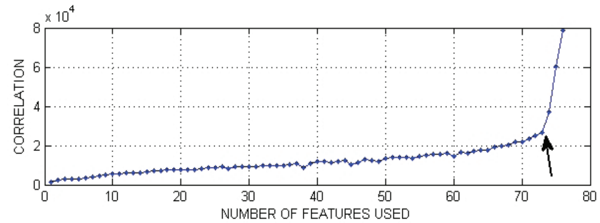

Fig. 3. **Correlation plot for selecting the optimum number of rank-ordered features for texture image classification, with the vertical arrow marking the heel point (with large gradient) in the curve.**

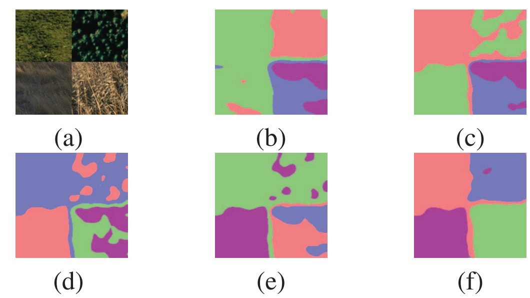

Fig. 4. **Segmentation results for texture image (a); obtained using (b) only Daubechies filter; (c) only DCT filter; (d) combination of Haar and DCT; all features (e) before feature selection and (f) after feature selection.**

that the ranked feature set not only assigns correct ranks to the features, but also provides a feature subset that maximizes the classification accuracy. The performance was obtained by averaging over 30 random observations. The dotted red curve shows the accuracy obtained using the ranked feature set and the horizontal baseline shows the accuracy obtained using the entire feature set. The time complexity of the proposed method of feature ranking is $O(D^3)$, which is similar to [9], and that of the feature selection algorithm is $O(D^2)$.

The texture images used for experiments are of size 256×256 created from VisTex texture database [11]. Feature extraction process using all the RGB color bands yields 76 features (4 from Daubechies, 36 from DCT and 36 from the combination of Haar and DCT filters). Fig. 6(b) shows the segmentation results using the 76 dimensional feature set of the texture images shown in Fig. 6(a). Fig. 6(c) shows the final segmentation results using the selected rank ordered features. DCT and DWT features have differing frequency selectively, producing non-identical (non-redundant) feature sets in terms of information for better discriminability. Table 2 gives the percentage accuracy of segmentation before and after feature selection. Segmentation accuracy is calculated based on the correct set of pixels classified in a region.

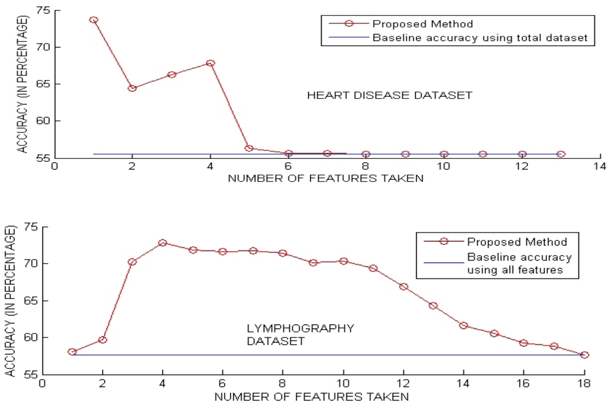

Fig. 5. **Change of classification accuracy by selecting increasing number of ranked features for fusion.**

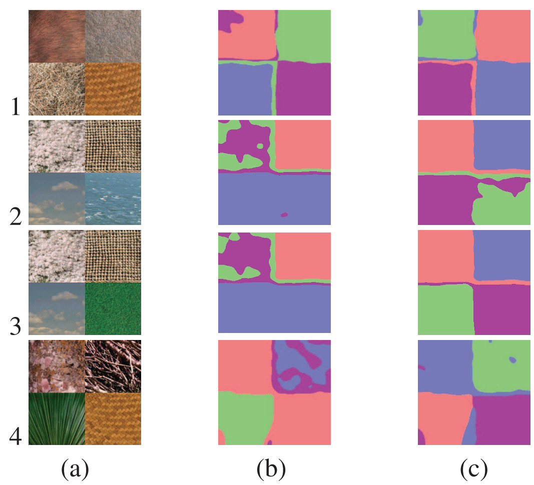

Fig. 6. **(a) Original texture images; Segmented results: (b) before feature selection; (c) after feature selection.**

5. CONCLUSION

In this paper, we have proposed an efficient method to select the best feature set for unsupervised feature selection. The method combines and selects the features using the proposed correlation criteria. Results show that the selection criteria removes the confusing set of features, improving the accuracy to a large extent. The selection process can be made faster by maintaining a lookup table containing pairwise feature correlation values and variance of each feature distribution instead of computing the value at every iteration. The method may fail in situations when the features have too much of redundancy or noise. Results can be improved by MRF based segmentation, or Graph cut, instead of using FCM.

Table 2. **Segmentation Accuracy (in %-age) for the texture images in Fig. 6.**

Image	1	2	3	4
After feature fusion	90.79	62.19	64.61	59.70
After feature selection	93.35	90.81	96.41	92.21

6. REFERENCES

[1] Trygve Randen and John Hakon Husoy, "Filtering for texture classification: A comparative study," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 4, pp. 291–310, 1999.

[2] Shivani Rao, Manika Puri, and Sukhendu Das, "Unsupervised segmentation of texture images using a combination of Gabor and wavelet features," in *Indian Conference on Computer Vision, Graphics and Image Processing*, 2004, pp. 370–375.

[3] Yindi Zhao, Liangpei Zhang, and Pingxiang Li, "Texture feature fusion for high resolution satellite image classification," in *International Conference on Computer Graphics, Imaging and Vision*, 2005, pp. 19–23.

[4] Huan Liu and Lei Yu, "Toward integrating feature selection algorithms for classification and clustering," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 4, pp. 491–502, 2005.

[5] Jian Yang, Jing yu Yang, David Zhang, and Jian feng Lu, "Feature fusion: parallel strategy vs. serial strategy," *Pattern Recognition*, vol. 36, no. 6, pp. 1369–1381, 2003.

[6] Shutao Li and Yaonan Wang, "Feature selection and fusion for texture classification," in *LNCS*, 2005, vol. 3497/2005.

[7] David A. Clausi and Huang Deng, "Design-based texture feature fusion using Gabor filters and co-occurance probabilities," *IEEE Transactions on Image Processing*, vol. 14, no. 7, pp. 925–936, 2005.

[8] Pabitra Mitra, C. A. Murthy, and Sankar K. Pal, "Unsupervised feature selection using feature similarity," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 3, pp. 301–312, 2002.

[9] Mark A. Hall, "Correlation-based feature selection for discrete and numeric class machine learning," in *International Conference on Machine Learning*, 2000, pp. 359–366.

[10] C.L. Blake and C.J. Merz, "UCI repository of machine learning databases," http://archive.ics.uci.edu/ml/.

[11] "Vistex database," http://vismod.media.mit.edu/vismod/imagery/VisionTexture/vistex.html.