A CLASSIFICATION FOR PANCHROMATIC IMAGERY BASED ON INDEPENDENT COMPONENT ANALYSIS

Ho-Young Lee
Dept. of Electronic Engineering, Sogang University
1 Shinsu-Dong, Mapo-Gu, Seoul, Korea, 121-742
lhy2502@eerobot1.sogang.ac.kr

Jun-Oh Park
Dept. of Electronic Engineering, Sogang University
1 Shinsu-Dong, Mapo-Gu, Seoul, Korea, 121-742
j3841108@hanmail.net

Kwae-Hi Lee
Dept. of Electronic Engineering, Sogang University
1 Shinsu-Dong, Mapo-Gu, Seoul, Korea, 121-742
khlee@sogang.ac.kr

Abstract: Independent Component Analysis (ICA) is used to generate ICA filter for computing feature vector for image window. Filters that have high discrimination power are selected to classify image from these ICA filters. Proposed classification algorithm is based on probability distribution of feature vector.

Keywords: ICA, Classification, Panchromatic imagery.

1. Introduction

The panchromatic imagery includes visual data such as edges, lines, and textures. For panchromatic image classification, methods using Co-occurrence Matrix, Fourier, DCT, Wavelet and PCA are used in many researches that have been carrying out to efficiently extract features from panchromatic image and reduce the dependency and correlation of the visual data. In this paper, we propose the method using ICA (Independent Component Analysis) as the method to extract features from panchromatic image. ICA removes the inter-related statistical dependency and transforms observed image signal to minimized linear combination [1]. ICA is suitable for analyzing image data that contains different shapes and sizes of edges each class.

The construction of this paper is as follow. In the second section, ICA algebraic concept is briefly introduced. In the third section, we explain the method of generating ICA filter from image windows. We also introduce the method to select ICA filters that have high discrimination and to classify feature vectors obtained from the selected filters. In the fourth section, the performance of proposed classification algorithm is shown. In the final chapter, we bring to the conclusion.

2. Independent Component Analysis

If independent source vector and observed random vector are $\mathbf{s} = [s_1, s_2, \cdots, s_n]^T$ and $\mathbf{x}_{obs} = [x_1, x_2, \cdots, x_m]^T$, the ICA linear model is as follow [2].

$$\mathbf{x}_{obs} = \mathbf{As}, \tag{1}$$

$$\mathbf{s} = \mathbf{W}\mathbf{x}_{obs}, \tag{2}$$

ICA is the method that assumes independent source vector $\mathbf{s}$, mixing matrix $\mathbf{A}$ ($m \times n$) or separating matrix $\mathbf{W}$ ($n \times m$) only from given $\mathbf{x}_{obs}$. In equation (1), $\mathbf{x}_{obs}$ is represented as the linear combination of independent source vector of factor s_i corresponded to each base vector $\mathbf{a}_i$ ($i = 1, \cdots, n$). Similarly in equation (2), once independent component s_i is generated from the matrix multiplication of the row vector of $\mathbf{W}$ and the $\mathbf{x}_{obs}$, $\mathbf{w}_i^T$ ($i = 1, \cdots, m$) becomes an ICA filter reflecting $\mathbf{x}_{obs}$ on the independent plane.

To assume an independent component, assumed value y (simply say $\mathbf{x}_{obs}$ to $\mathbf{x}$) can be expressed as follow [1].

$$y = \mathbf{w}_i^T \mathbf{x} = \mathbf{w}_i^T \mathbf{As} = \mathbf{z}^T \mathbf{s}, \tag{3}$$

$$\mathbf{z} = \mathbf{A}^\mathbf{T} \mathbf{w}_i,$$

The assumed independent component y can be represented as the linear combination of weight vector $\mathbf{z}^T$ and originally included independent component s_i. The purpose of ICA is making $\mathbf{z}^T$ to have a component, which is not a zero, as maximizing the non-gaussianity of $\mathbf{w}_i^T \mathbf{x}$. Here, $\mathbf{w}_i^T \mathbf{x} = \mathbf{z}^T \mathbf{s}$ would be corresponded to the one of included independent components.

Fast ICA algorithm used in this paper makes use of negentropy to measure non-gaussian [3].

3. Proposed Classification Algorithm

The proposed classification method consists of four major steps. Those are preprocessing, ICA filter extraction, filter selection and image classification.

As a preprocessing step, we first carry sharpening out to emphasize contrast between neighboring pixels and maximize texture [4]. Next, image is normalized and made zero mean. As the second step, the training data obtained from the preprocessed image is subtracted its local mean and then ICA is performed to generate ICA filter. We extracted 50 filters by reducing dimension with PCA. Figure 1 shows example of extracted ICA filters.

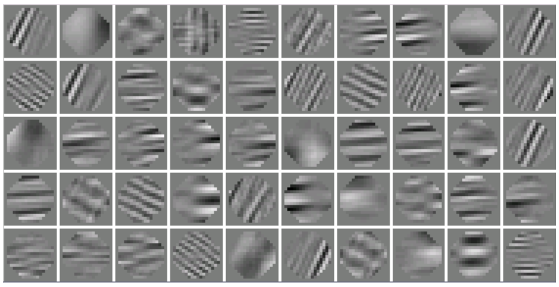

Figure 1. Example of extracted ICA filter (18 x 18)

After we calculate feature vector (here independent source vector) from equation (2) using extracted filters, we normalize the feature values created from each filter to values between 0 and 255. We use criterion function J to select a filter that has good discrimination power [5].

$$J = \frac{\left| mean(s_i^{c_1}) - mean(s_i^{c_2}) \right|}{var(s_i^{c_1})^2 + var(s_i^{c_2})^2}, \tag{4}$$

where $s_i^{c_1}$ is the feature values of class c_1 calculated through filter i, $mean(s_i^{c_1})$ and $var(s_i^{c_1})$ is the mean and variance of $s_i^{c_1}$. If J has values big enough comparing to some pair of classes $\{c_1, c_2\}$, we consider filter i as a filter that has good discrimination power, and so we use them for classification. Through above process, we select 5 filters.

As a final step, we evaluate the probability distribution of feature values obtained from each filter for each class by using training data. We extract feature values with each selected filter from the entire image. Then feature map that is composed of feature values obtained from each filter is created as much as filters. Figure 2 shows feature maps generated by each filter.

To consider both center feature value and values around it, we use window (19×19). We calculate the distribution of feature values within window for each feature map and then compare it with probability distribution obtained from training data.

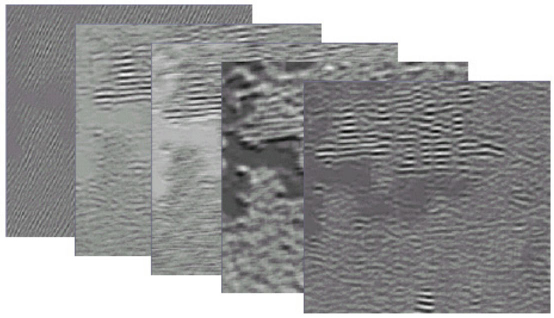

Figure 2. Feature maps generated by each filter

To calculate the similarity, we used the summation of absolute difference and find a class that has a minimum value for each feature map. Finally, we select a class as a conformed class if it shows most of filters.

For area that isn't classified, we classify them with pixel unit. We first find a class that has maximum value in product of probability of feature values with same position for each feature map. We finally assigned a class occupying majority within window to conformed class.

4. Experimental Result

The image used in the experiment is an aerial image of Seoul province with 4-m resolution. The size of the image is 300×300. We obtained 100 sample data from each class in the image. Then we rotated each sample image in 45 degree increment and so get 800 training data set for each class [6].

The experiment image in Figure 3 composed of 3 classes that are mountain area, residential area with low buildings, and residential area with high buildings. We classify the image with 3 classes. As a result, mountain area, residential area with low building, and residential area with high building are classified as class 1, class 2, and class 3 as shown in Figure 3. To evaluate the performance of classification, we verified the accuracy of assigned class by randomly selecting 600 coordinates in the original image. As a result, we obtained 91.2% accuracy as shown in Table 1.

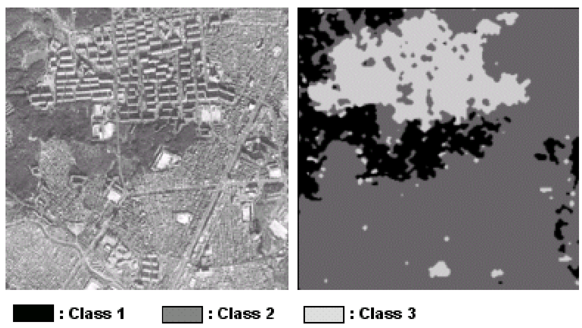

Figure 3. Original image and classified images

Table 1. Accuracy for classification

Class	Class1	Class2	Class3
Num of sample	128	309	163
Accuracy(%)	91.6	92.3	89.7

5. Conclusion

We obtained feature values by analyzing edges, lines, and texture information of the panchromatic image with ICA and applied the algorithm to classify those feature values. In our experiment, we obtained about 91% accuracy in classification with this method.

In ICA filter, localization is guaranteed in spatial frequency and orientation and we are able to create filters with more complex frequency response comparing to other methods [6]. Therefore, we conclude that ICA is very useful in extracting information from image and also has good performance to generate feature values for efficient classification.

References

[1] A. Hyvarinen, 1999. Survey on Independent Component Analysis, *Neural computing surveys*, vol. 2, pp. 94-128.

[2] A. Hyvarinen, J. Karhunen, 2001. *Independent Component Analysis*, John wiley&Sons Inc.

[3] A. Hyvarinen, 1999. Fast and robust fixed-point algorithms for independent component analysis, *in Trans. IEEE Neural Networks*, vol. 10, pp. 623-634.

[4] R. Gonsalez, R. E. Woods, 2002. *Digital image processing*, Prentice Hall Inc.

[5] Richard O. Duda, Peter E. Hart, David G. Stork, 2001. *Pattern Classification*, John wiley&Sons Inc.

[6] Robert Jenssen, Torbjorn Eltoft, 2003. ICA FILTER BANK FOR SEGMENTATION OF TEXTURED IMAGES, *ICA2003*, Japan, pp.827-832.