SCENE IMAGE RECOGNITION BASED ON THE SEQUENCE OF LOCAL IMAGE VECTORS REPRESENTED BY ORIENTED EDGES

Shigetaka Morikawa [1] *and Tadashi Shibata* [2]

[1] Department of Frontier Informatics,
[2] Department of Electrical Engineering and Information Systems,
The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo, 113-8656, Japan

ABSTRACT

A new feature description algorithm has been developed for scene image recognition. Since scene images have a large variation in the same scene category, the sequence representation using local image vectors has been explored in this work. The structural features present in scene images are enhanced by using oriented edges to represent local images. As a result, it has become possible to handle the large variation in scene images robustly. Using the test images from seven categories, extensive experiments were carried out. In some of the sample images, test images were resized by either changing the resolution or cutting off at the peripheral area, and evaluated the robustness of the algorithm. The results were compared to Scale Invariant Feature Transform (SIFT) [1], and robustness of the proposed method against the variation in scene images has been confirmed.

Index Terms— Feature extraction, edge detection, image vector sequence, dynamic programming, scene image recognition

1. INTRODUCTION

Digital image data are increasing explosively today by the evolution of image hosting websites and the popularization of digital cameras. As a result, people now save and access a large number of images. To organize and search for images for a variety of purposes, an effective method is required. To know what scene the image represents is very important for image processing. For example, Lazebnik et al. in [2] describe "if the image, based on its global description, is likely to be a highway, we have a high probability of finding a car, but not a toaster." Then scene image recognition is very useful for image understanding and object detection. Scene images are composed of various components (e.g. tree, road and building). In each image, the components take different shapes and appear on different locations. Even in the same scene category, scene images have a large variation. Therefore, we believe this is the most difficult issue in scene image recognition.

In early years, for scene image recognition, a vector representing the global feature is directly generated from low level features in an image. The vector includes less information about the locations of local features. Therefore, the representation can handle variations in images. About the global feature, Vailaya et al. [3] use histogram by counting edges in various angles. Pass et al. [4] use color vector with coherence of contiguous pixels. But there are limitations in such vector representations. They are too simple to handle a number of complicated scene categories

In recent years, intermediate representations are generated from local region in an entire image. The representation includes characteristic parts of the image. And they are assembled to enhance the characteristic features of the scene image or extract semantic meaning from the image. Such representations include more information about complicated scene images and, at the same time, have reduced the influence from variation. About the intermediate representation, Sivic and Zissermanl [5] proposed to use affine invariant regions represented by SIFT as words in conjunction with vector quantization. This framework known as bag-of-features makes it possible to treat scene images just like the text of sentences and apply a number of algorithms for text categorization to image classification problems. They use "term frequency–inverse" weighting to enhance characteristic features. Following this work, such framework has been applied widely to scene recognition. For example, Quelhas et al. [6] generate term document matrix from images and apply Probabilistic Latent Semantic Analysis (PLSA) to images for extraction of semantic meaning. Aside from the framework, Oliva and Torralba [7] use spatial frequency of local region and PCA for dimension reduction.

For scene image recognition, there are various features like color, texture and edge. In particular, recently SIFT as gradient-based feature has been widely used. Regarding the gradient based features, Dalal and Triggs [8] propose Histogram of Oriented Gradient (HOG) and Yagi et al. [9] propose Projected Principal-Edge Distribution (PPED). HOG represents the distribution of gradient at a local region and PPED enhances structural feature by projection of oriented edges. And SIFT finds only characteristic points and ignores unnecessary parts in an image. Therefore SIFT is ro-

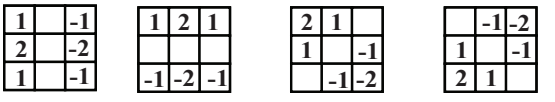


Fig. 1. Four oriented edge filters to identify edge candidate locations as peaks in the gradient about each orientation

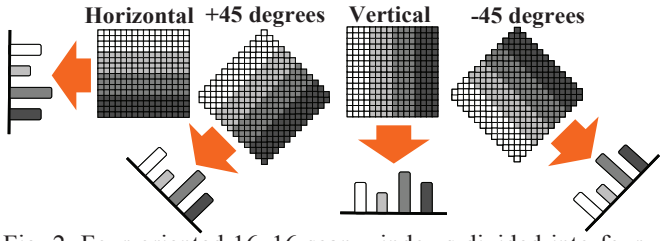


Fig. 2. Four oriented 16x16 scan windows divided into four regions to project and bundle oriented edges into four bins.

bust against the variation of images. On the other hand, Ratan et al. [10] use extensible templates to respond to the variation for object detection. The templates absorb the variation in object images by using DP matching.

In this paper, we propose a new feature description algorithm as the intermediate representation of scene images. Unlike the keypoints employed in SIFT, we focus on the structural features in an image, i.e. oriented edges. Four oriented edges have been chosen and employed to represent local images. Therefore, in the present method, a scene image is represented as plural of sequences of local images each represented by oriented edges. DP matching was employed for similarity evaluation between sequences. We divide an image into many sequences as mentioned above and apply PLSA to the sequences as in the work of [6]. For experiments, we vary the resolution and cut off a part of test images and compare our method to SIFT. The results show that the performance of our method is superior to SIFT and robust against the variation.

In this paper, in section 2, we explain the proposed method in detail. Section 3 describes the experiments and Section 4 discusses the experimental results. Finally in section 5, conclusions are given.

.

2. ALGORITHM

The algorithm starts by extracting oriented edges from an input image and local images are represented by the spatial distribution of edges in respective areas. Then the entire image is converted to a collection of local image vectors. The essence of the algorithm is to view an input image as if it were a document of sentences in which local image vectors play the role of "letters". Then we need to identify a "word" composed of letters. Therefore we cut the entire image into thin threads (in horizontal direction, for instance), and then the threads are divided into meaningful pieces, which correspond to "words" in a sentence. After a scene image is decomposed into words, the strategy as described

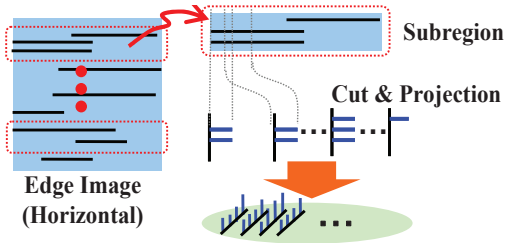


Fig. 3. An extraction of local image vectors from thread in a horizontal edge image.

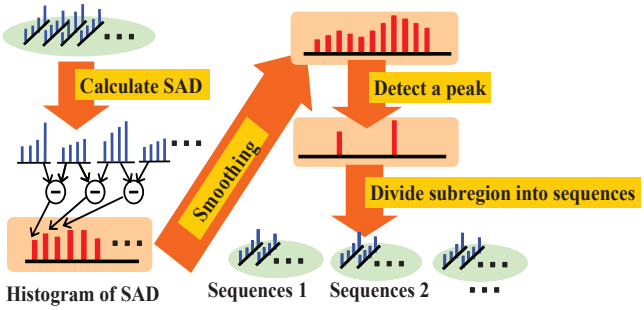


Fig. 4. A separation of local image vectors in a thread to generate sequences.

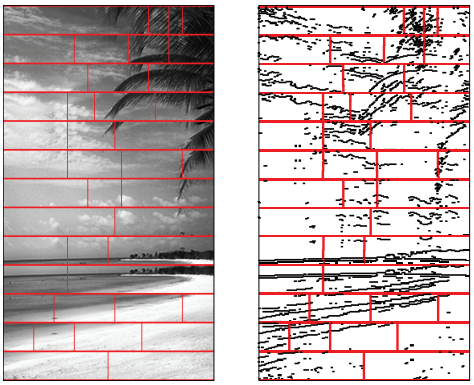


Fig. 5. Results of a separation of local image vectors about horizontal

in [6] is used. A document vector is generated from words in an image and converted to a topic vector by using PLSA. The detail is described in the follow.

In order to extract thin edge lines, oriented edges in four directions (horizontal, vertical, +45° and -45°) are detected as in the following. Using Sobel filters (left two filters in Fig.1), horizontal and vertical gradients are calculated after a 3x3 median filter for noise reduction. Edge candidate locations are identified as peaks in the gradient where diagonal gradients are also evaluated using the right two filters in Fig.1. Those edge candidates having gradients that exceed a certain threshold are all retained as edges, and thus four edge maps are produced.

Horizontal edge map is cut into threads (subregions) as shown Fig. 3 with a half-pitch overlapped. Then a square window scans the thread from one end to the other. Producing edge distribution vectors at equal-space locations

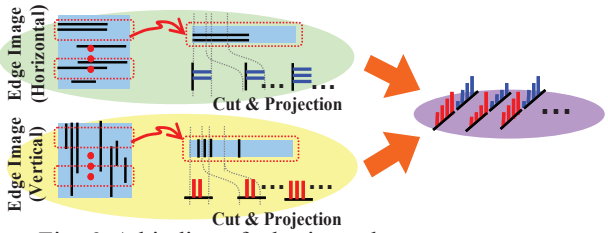


Fig. 6. A binding of a horizontal vector sequences and an additional vertical vector sequences.

(space=1/4 of the window width). The vector is, in this case, is a four dimension vector produced as shown in Fig.2 (horizontal). In this work, the thread was produced in four directions separately in similar manners, where edge distribution vectors are produced according to each direction as shown in Fig.2.

How to divide the subregion (thread) into meaningful pieces is explained below. Manhattan distance is calculated between neighboring local image vectors and the series of distance data is produced as in Fig. 4. After smoothing, the peak positions are detected, which serves as partitioning boundaries. Each partitioned area is made up of similar local images, and is call a "sequence" hereafter. Fig. 5 shows the result of a separation of local image vectors about horizontal edge on gray scale and edge image.

After sequence generation, we use the same strategy as [6]. A document vector is generated from sequences in an image and converted to a topic vector by using PLSA. We use DP matching to calculate similarity between the sequences and generate a codebook by K-medoids. Four document vectors are generated from an image and bundled into one vector. the vector is used for PLSA.

In order to enhance the feature representation ability of horizontal vector sequences, vertical edge information is included as vector elements as illustrated in Fig. 6. Horizontal threads are produced in the vertical edge map and they are partitioned at the same locations of the horizontal subregion. The horizontal edge distribution in each local area (see Fig.2) is bundled with the vertical edge distribution to yield an enhance representation (8-dimention vector). In this way, the additional orthogonal oriented edge information is added to the other three oriented vector sequences like horizontal vector sequences.

3. EXPERIMENT

The performance of the proposed method was evaluated by varying the parameters. We configured seven categories of scene that were Beach, Field, Forest, Highway, Mountain, Street and Tall Building. We selected 100 images at each category from the database used in [7] by Olive et al. 100 images were divided into 50 images for training and others for test. We converted them into 256x256 pixel gray scale images.

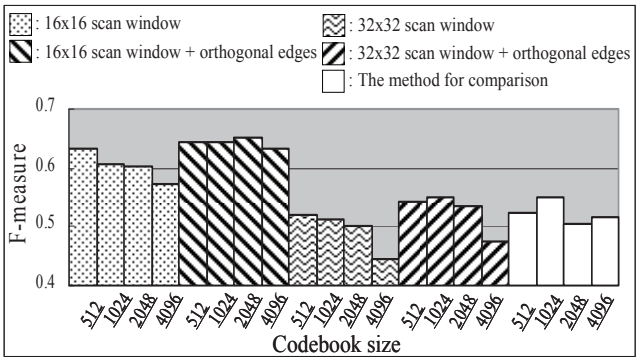
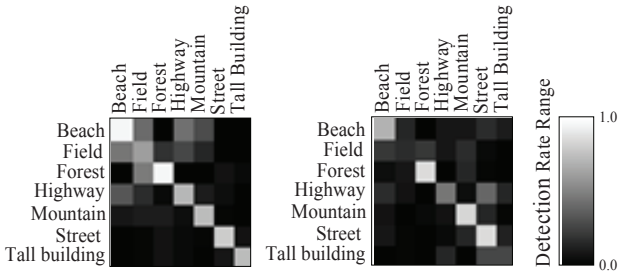


Fig. 7. Results of the best performance by each method in various codebook size.

Table 1. Comparison of two methods in the best performance.

	Our proposal method	The method for comparison
Scan window	16x16 + orthogonal edges	none
Codebook size	2048	1024
Number of topics	16	16
Threshold	0.61	0.56
F-measure	0.65	0.55



Our proposal method The method for comparison

Fig. 8. Confusion matrixes in the best performance.

As for parameters, the number of dimensions of the topic vector was 8, 16, 32, 64 and 128. A codebook size to generate the document vector by vector quantization was 512, 1024, 2048 and 4096. That meant a codebook size at each edge orientation was 128, 256, 512 and 1024. Scan window size was 16x16 and 32x32. About sequence, we used two kinds of them. One was generated by only each oriented edges. The other was generated by each oriented edges and additional orthogonal oriented edges.

A codebook was generated by K-medoids from 10 training images at each category. About PLSA, T. Hofmann [11] had proposed the tempered EM. But we simply used the EM algorithm. Each test image was compared to every training image by template matching. And the result was judged by each category. A threshold of template matching was swept to find the best performance.

We compared our proposed method to the same strategy as [6]. They had used SIFT and PLSA. To generate SFIT vector, we used binaries provided at [12]. The two methods were compared about the best performance. We made additional experiments. We used the parameters that showed the best performance in our experiment and retained training

Table 2. F-measure for various test images

	Our proposal method	The method for comparison
Resize 50%	0.54	0.51
Resize 200%	0.52	0.51
Cut 50% (vertical)	0.58	0.50
Cut 50% (horizontal)	0.54	0.52

images without change and varied test images. We resized test images to 50 % and 200%. Or we cut total 50% of test images from the both sides of either horizontal or vertical orientation.

4. RESULTS AND DISCUSSION

Fig. 7 shows the performances of our proposed method and the method for comparison. About each codebook size, the performances display the highest F-measure value when a threshold is swept in template matching at various numbers of topics. The performance of small scan window is better than large one. And totally our method with additional orthogonal edges is superior to that without the information. In the best performances of our method and the method for comparison, Table 1 shows the F-measure values and the parameters, and Figure 8 shows the confusion matrixes. In Table 1, our method is superior to the method for comparison. In Fig. 8, the confusion matrixes show that about beach, field and highway, our method does not work well and about field, highway and tall building, the method for comparison does not work well.

Regardless of the scan window size, the number of dimension of local image vector is same. And the scan window extends in two directions. But directional edges extend in only one direction. Therefore the large scan window tends to be more sensitive to noise. That degrades the performance. On the other hand, the information of additional orthogonal edge is useful to improve the performance. About beach, field and highway, they have similar structures as a whole and our method enhances the structural features. Therefore our method does not discriminate them well. On the other hand, about field, highway and tall building, the method for comparison does not work well. About field, it seems that there are few characteristic parts in the images to discriminate scene. Between highway and street, there are common structures like road and different structures like the presence of buildings. Between tall building and street, there are common structures like buildings and different structures like the presence of pavements. About highway and tall building, the methods for comparison does not discriminate them from street. But by using structural features, our method discriminates them from street.

Table 2 shows the performances when training images are changed variously. About 200% resized test image, our proposed method is slightly superior to the method for comparison. As a whole, the performance of our method is superior to the method for comparison.

About resized test images. Our method is superior to the method for comparison. But our method does not respond to multi-resolution. On the other hand, SFIT in the method for comparison responds to that. If test images are resized extremely, probably our method will be inferior to the method for comparison. But, about a defect of a scene image, the performance of our method is superior to the method for comparison too. Our method uses the sequence representation and calculates similarity between them by DP matching. That means our method has the effect of interpolating and is robust to the variation of images.

5. CONCLUSIONS

In this paper, we propose a new feature description algorithm for scene image recognition. Scene images have a large variation in the same scene category. Our proposed method is the sequence representation. The method enhances the structural features by using oriented edges and responds to the variation of images by using the representation.

In the experiments about a defect of a scene image, we show that our method is superior to the method for comparison that uses SIFT. By the results, the robustness of our method against the variation of scene images is confirmed.

6. REFERENCES

[1] D. Lowe, “Distinctive image features from scaleinvariant keypoints”, IJCV, pp. 91-110, 2004.

[2] S. Lazebnik, C. Schmid, and J. Ponce, “Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories,” CVPR, June 2006.

[3] A. Vailaya, A. Jain and H.Zhang, “On image classification: City images vs. landscapes,” Pattern Recognition,” vol.31, pp.1921–1935, 1998.

[4]G. Pass, R. Zabih, and J. Miller, “Comparing images using color coherence vectors,” 4th ACM Conference on Multimedia, Boston, 1996

[5] J. Sivic and A. Zisserman, “Video google: A text retrieval approach to object matching in videos,” ICCV, pp. 1470-1477, 2003.

[6] P. Quelhas, F. Monay, J. Odobez, D. Gatica-Perez, T.Tuytelaars, and L. Van Gool, “Modeling scenes with local descriptors and latent aspects,” ICCV, Beijing, China, 2005.

[7] A. Oliva and A.Torralba, “Modeling the shape of the scene: A holistic representation of the spatialenvelope,” IJCV, pp. 145-175, 2001.

[8] N. Dalal, B. Triggs, “Histograms of oriented gradients for human detection,” CVPR, pp. 886-893, 2005.

[9] M. Yagi, M. Adachi, and T. Shibata, “A Hardware-Friendly Soft-Computing Algorithm for Image Recognition,” EUSIPCO, pp. 729–732, 2000.

[10] A. L. Ratan, W. E. L. Grimson, and I. William M. Wells, bject detection and localization by dynamic template warping”, CVPR, pp. 634-640, 1998.

[11] T. Hofmann, “Unsupervised learning by probabilistic latent semantic analysis,” Mach. Learning, pp. 177-196, 2001

[12] http://www.cs.ubc.ca/~lowe/keypoints