Classification of Synthetic and Real Images Using Pattern Features

Myeong Hui Ha [1], Hyun jun Choi [1], Min Kook Choi [2], and Sang Chul Lee [3]

[1,2,3]Department of Computer Science, Inha University, Incheon, South Korea

Abstract— **Automatic classification of motion pictures has many application areas, and one prominent case is image search based on a user query. In this particular application, the users intention for a query is difficult to identify in the semantic level because image search algorithm generally exploits only features of an image. In order to address this issue, we propose in this paper an automatic image classification algorithm applicable to synthetic and real motion pictures so that the users intention could be fully reflected at image search and classification. Feature-based or edge-based histogram descriptors produce poor results when applied to the classification of synthetic and real images because this is inter-class classification. The algorithm proposed in this paper obtains feature vectors of color distribution pattern in motion pictures to classify synthetic and real images. In our experiment, the proposed algorithm was able to classify images with an accuracy of around 74%.**

I. INTRODUCTION

The majority of image classification algorithms are intra-class classification methods and use low-level visual features or text features to distinguish images of different categories, such as shots of mountains, beach, sunset scenery, or urban landscape [2].

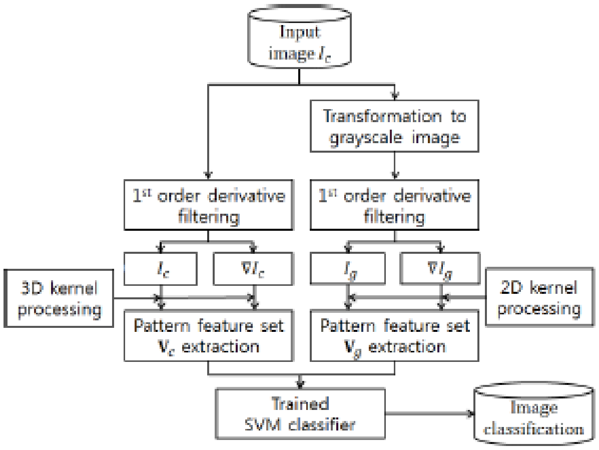
Existing work [1, 2] generally do not distinguish similar images with different semantic meanings. Similar images are put into the same categories, and it is hard to provide service that fit user query that has different semantic intentions. Hence, it is necessary to implement a classification method tailored for semantic classification. One example of classification within the same category is distinguishing synthetic images and real images. Lienhart and Hertman [3] was among the first to explore this topic; their algorithm distinguished graphical images and photo-like images, which were further classified into true photos and ray-traced/rendered images. Other image classification algorithms such as SIFT (scale-invariant feature transform), SURF (speeded-up robust features), BRISK (binary robust invariant scalable keypoints), and GLOH (gradient location and orientation histogram) are feature-based descriptors [4, 6, 7, 10]; HoG (histogram of oriented gradients) exploits edge histogram-based features [5]. Existing feature descriptors show great performance for distinguishing images of different categories but not for images within the same category. This weakness is caused by the fact that image descriptors are based on object edges. Here have been attempts to improve on this weakness by analyzing images through CBIR (content-based image retrieval) [8] and Garbarinos work [9] that exploits textual feature. In this paper, we propose an algorithm for synthetic and real images by combining widely used feature descriptors with color patterns and feature distribution analysis. We chose four image classes of apple, human face, landscape, and buildings, all of which can be processed with binary classification. With the proposed algorithm, it is possible to implement a search/classification system that can meet user query at the semantic level.

II. Pattern Feature Extraction

In this paper, we propose a pattern-based feature extraction algorithm that classifies synthetic and real images. Unlike feature descriptors and histogram-based features, pattern features of images consider the distribution of pixel strengths

Myeong Hui Ha email address: mkchoi@inha.edu
Hyun jun Choi email address: mkchoi@inha.edu
Min Kook Choi email address: mkchoi@inha.edu
Sang Chul Lee email address: mkchoi@inha.edu

within local regions. Such pattern data is based on the type of color distribution and local texture data, and this produces improved



Input image I_c
Transformation to grayscale image
1st order derivative filtering
1st order derivative filtering
I_c
∇I_c
I_g
∇I_g
3D kernel processing
2D kernel processing
Pattern feature set $\mathbf{V}_c$ extraction
Pattern feature set $\mathbf{V}_g$ extraction
Trained SVM classifier
Image classification

FIG 1: algorithm for classifying real and synthetic image

A. Two-dimensional Pattern Extraction

When the input images has been entered as an $m \times n$ tensor I_c∈ N$^{m\times n\times 3}$, then converting to black-and-white color scheme yields I_g∈ N$^{m\times n}$. To find the pattern vector V_g for I_g∈ N$^{m\times n}$ a kernel array set $f_w \in \{f_{w,1}, f_{w,2}, \dots, f_{w,6}\}$ is defined. Here, w is the parameter for the size of the kernel and it can be in four different forms. Equation (1) is an example when w=3 for 2-dimensional pattern extraction kernel.

$$f_{3,1}=\begin{bmatrix}-1 & 1 & -1\\ 1 & -1 & 1\\ -1 & 1 & -1\end{bmatrix} \mathrm{f}_{3,2}=\begin{bmatrix}-1 & 1 & -1\\ 1 & -1 & 1\\ -1 & 1 & -1\end{bmatrix}$$
$$f_{3,3}=\begin{bmatrix}-1 & 1 & -1\\ 1 & -1 & 1\\ -1 & 1 & -1\end{bmatrix} \mathrm{f}_{3,4}=\begin{bmatrix}-1 & -2 & 0\\ -2 & 0 & 2\\ 0 & 2 & 1\end{bmatrix}$$
$$f_{3,5}=\begin{bmatrix}0 & -1 & -1\\ 1 & 0 & -1\\ 1 & 1 & 0\end{bmatrix} \mathrm{f}_{3,6}=\begin{bmatrix}0 & -2 & -1\\ 2 & 0 & -2\\ 1 & 2 & 0\end{bmatrix} (1)$$

In order to reflect a wide range of characteristics in the images, first-order differential image ∇I_g for the black-and-white image I_g is calculated. Let the first-order differential images of I_g in the horizontal and vertical directions be ($I_{g,x}$,$I_{g,y}$)then the change of pixel at each pixel coordinates (u, v) for ($I_{g,x}$,$I_{g,y}$) is defined as follows:

$$I_{g,x}(u,v)\text{=}I_g(u,v)\text{-}I_g(u+1,v)$$
$$I_{g,x}(u,v)\text{=}I_g(u,v)\text{-}I_g(u,v+1)$$

Pixel strength at coordinates (u, v)is denoted by $I_g(u,v)$ Here, the first-order differential image can be defined as:

$$\nabla I_g = I_{g,x}\text{+}I_{g,y}$$

In order to obtain pattern features from two images of I_g and ∇I_g, f_w is applied, which can be in one of three different forms. The first form is applied to the i th element of f_w, or $f_{w,i}$ as a convolution calculation:

$$b_{1,i} = I_g * f_{w,i} \text{ (2)}$$
$$b_{2,i} = \nabla I_g * f_{w,i} \text{ (3)}$$

The term b_i is the convolution calculation result for the i th kernel defined within the range of $1 \le i \le 6$. Following Equations (2) and (3), the feature vector $v_1(u,v)\text{=}\left[b_{1,1}, \dots, b_{1,s}\text{,}b_{2,1}, \dots, b_{2,s}\right]^\top$ is extracted on the pixel at coordinates (u,v). With the type of kernel and input image, we obtain s=6. Secondly, f_w is used for extracting the statistical parameters for the i th element $f_{w,i}$. The term $b_{3,i}(u,v)$ is defined as the weighted average for pixel I_g on coordinates (u,v) in the kernel matrix $f_{w,i}$. Next, the term $b_{4,i}(u,v)$ is the weighted average for pixel ∇I_g on coordinates (u,v) in the kernel matrix $f_{w,i}$, and it is used for extracting feature vector $v_2(u,v)$. The third form of pattern extraction for the i th element $f_{w,i}$ in f_w also is a statistical parametric form. The term $b_{5,i}(u,v)$ is defined as the weighted standard deviation for the pixel I_g on the kernel matrix I_g on coordinates (u,v). We also define $b_{4,i}(u,v)$ as the weighted standard deviation for the pixel ∇I_g, and this is used to extract the feature vector $v_3(u,v)$ In short, the algorithm performs $I_c \rightarrow I_g$ black-and-white conversion for the 2-dimensional input image I_c and produces the first-order differential image ∇I_g in order to extract pattern features.

With set operations of f_w kernel matrix on the converted input image I_g and ∇I_g, pattern vector set V $\in$ {v$_1$,v$_2$, …,v$_k$} is obtained, where k is the total number of feature vectors for the input image I_c and the i th element of set V is v_i =$\left[b_1\text{,}b_2, \dots, b_s\right]^\top$. The letter s denotes the dimension of each feature vector. If we follow the kernel matrix of Equation (1) in this case, a feature vector set V with k=3 and s=12 is produced.

B. Three-dimensional Pattern Feature Extraction

Three-dimensional pattern feature extraction is performed on the color input image I_c. Following the same principles as in the 2-dimensional pattern feature extraction, we define a 3-dimensional kernel trifocal tensor set $f'_q \in \{f'_{q,1}, f'_{q,2}, \ldots, f'_{q,6}\}$, which is an extension from the 2-dimensional kernel in Equation (1), in order to find the pattern vector set V' for the input image I_c. The extended 3-dimensional kernel tensor $f'_{q,1}$ as an extension from the kernel matrix $f_{w,1}$ in Equation (1) is defined as follows:

$$f'_{q,1,1}= \begin{bmatrix} 1 & -1 & 1 \\ -1 & 1 & -1 \\ 1 & -1 & 1 \end{bmatrix} \text{f'}_{q,1,2}= \begin{bmatrix} -1 & 1 & -1 \\ 1 & -1 & 1 \\ -1 & 1 & -1 \end{bmatrix}$$

$$f'_{q,1,3}= \begin{bmatrix} 1 & -1 & 1 \\ -1 & 1 & -1 \\ 1 & -1 & 1 \end{bmatrix} \text{ (2)}$$

The index i in the trifocal tensor $f'_{q,1,i}$ denotes the matrix plane within the tensor and it consists of extended $f_w \rightarrow f'_q$ as in Equation (1). The first-order differential image ∇I_c can be defined as follows for each color plane $I_{c,r}$, $I_{c,g}$, $I_{c,b}$ in the color image I_c:

$$\nabla I_{c,r}=I_{c,r,x}+I_{c,r,y}$$
$$\nabla I_{c,g}=I_{c,g,x}+I_{c,g,y}$$
$$\nabla I_{c,b}=I_{c,b,x}+I_{c,b,y}$$

Here, $\nabla I_{c,r}$, $\nabla I_{c,g}$,and $\nabla I_{c,b}$ denote the first-order differential image on each of the RGB color planes. The equation to extract a pattern feature vector set V' from 3-dimensional image I_c and ∇I_c extends the equation for 2-dimensional image. Similar to the 2-dimensional case, $v'_4(u,v) = [\text{b}_{7,1}, \ldots, b_{7,s}, b_{8,1}, \ldots, b_{8,s}]^\top$ is found with $b_{7,i}$ and $b_{8,i}$; $v'_5(u,v) = [b_{9,1}, \ldots, b_{9,s}, b_{10,1}, \ldots, b_{10,s}]^\top$ with $b_{9,i}$ and $b_{10,i}$; $v'_6(u,v) = [b_{11,1}, \ldots, b_{11,s}, b_{12,1}, \ldots, b_{12,s}]^\top$ with $b_{11,i}$ and $b_{12,i}$. Following the type of kernel and input image, we obtain s = 12. Three-dimensional input image I_c is processed to obtain ∇I_c similarly as in processing 2-dimensional pattern features, and a pattern vector set V' $\in$ v'_1,v'_2,...,v'_k is extracted by performing set operations with f'_q kernel tensor set on the input image I_c and ∇I_c. The i th element in V', is defined as $v'_i = [b_1, b_2, \ldots, b_s]^\top$; following the kernel matrix of Equation (1), we obtain a feature vector set V' where k=3 and s=12.

III. Implementation of Pattern Feature-based Classifier

From the feature vector sets V and V' extracted from input image dataset, we implemented an SVM classifier, which is defined as a Lagrange optimization problem as in Equation (3), which finds a hyperplane with maximum margin:

$$\max_\alpha L(\alpha) =$$

$$\sum_{i=1}^{n} \alpha_i - \frac{1}{2\lambda} \sum_{i=1}^{n} \sum_{j=1}^{n} (\alpha_i, y_i)(\alpha_j, y_j) K(b_i, b_j)$$

$$\text{s.t. } y\alpha = 0, 0 \le \alpha_i \le \tfrac{1}{n} \text{ (3)}$$

The class label variable y in this paper satisfies $y \in \{1, 1\}$; b is the feature vector of extracted pattern features.

IV. Experiment and Analysis

To test the performance of our proposed algorithm, we downloaded from the Internet images of apples, people, landscapes, and buildings to distinguish synthetic and real images. In [Figure 2], examples of the images selected are shown.

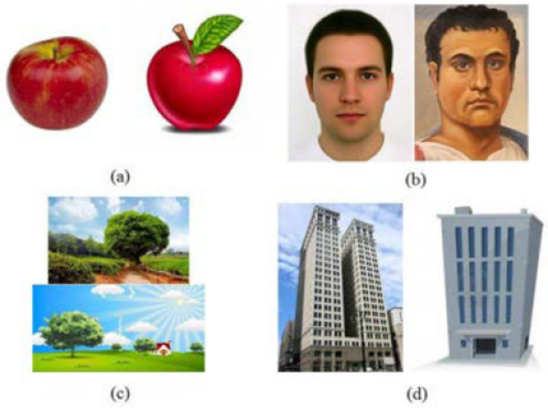


FIG 2: 4 kinds of synthetic/real images for image classification (a) apple, (b) face, (c) landscape, and (d) building

For algorithm evaluation, we collected 100 files for synthetic and real images each and performed 5-fold cross-validation. We trained our classifier with 80 synthetic and 80 real images for each category and tested the classifier with 20 synthetic and 20 real images. We made 21 different compositions of training and test images for experiment and calculated precision and recall values:

Here, tp, fp, and fn denote true positive, false positive, and false negative, respectively. In [Table 1], precision and recall results for each category are shown.

$$\text{precision} = \frac{tp}{tp+fp}$$
$$\text{recall} = \frac{tp}{tp+fn}$$

apple	real	synthetic	precision
real	144	66	0.686
synthetic	137	73	0.348
recall	0.512	0.525	0.517
			0.519

face	real	synthetic	precision
real	189	21	0.9
synthetic	89	121	0.576
recall	0.68	0.852	0.738
			0.766

landscape	real	synthetic	precision
real	127	83	0.605
synthetic	139	71	0.338
recall	0.477	0.461	0.471
			0.469

building	real	synthetic	precision
real	169	41	0.804
synthetic	124	83	0.41
recall	0.577	0.677	0.607
			0.627

TABLE I: precision-recall by category.

V. Conclusion

In this paper, we proposed a classification method for images in the same category. Exploiting pattern features (texture) showed improved classification accuracy over other classification methods based on points, edges. In our future work, we will advance from the current form of kernels and extract more detailed feature extraction and different types of statistics data for localized areas in images.

VI. Acknowledgements

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIP) (No. 10047078).

References

[1] John R Smith and Shih-Fu Chang. “Image and video search engine for the world wide web”. In: *Storage and Retrieval for Image and Video Databases V*. Vol. 3022. International Society for Optics and Photonics. 1997, pp. 84–95.

[2] Aditya Vailaya et al. “Bayesian framework for semantic classification of outdoor vacation images”. In: *Storage and Retrieval for Image and Video Databases VII*. Vol. 3656. International Society for Optics and Photonics. 1998, pp. 415–426.

[3] Rainer W Lienhart and Alexander Hartmann. “Classifying images on the web automatically”. In: *Journal of electronic imaging* 11.4 (2002), pp. 445–455.

[4] David G Lowe. “Distinctive image features from scale-invariant keypoints”. In: *International journal of computer vision* 60.2 (2004), pp. 91–110.

[5] Navneet Dalal and Bill Triggs. “Histograms of oriented gradients for human detection”. In: 2005.

[6] Krystian Mikolajczyk and Cordelia Schmid. “A performance evaluation of local descriptors”. In: (2005).

[7] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. “Surf: Speeded up robust features”. In: *European conference on computer vision*. Springer. 2006, pp. 404–417.

[8] Fei Wang and Min-Yen Kan. “NPIC: Hierarchical synthetic image classification using image search and generic features”. In: *International Conference on Image and Video Retrieval*. Springer. 2006, pp. 473–482.

[9] Maurizio Garbarino. *Automatic classification of natural and synthetic images*. Skolan för datavetenskap och kommunikation, Kungliga Tekniska högskolan, 2008.

[10] Stefan Leutenegger, Margarita Chli, and Roland Siegwart. “BRISK: Binary robust invariant scalable keypoints”. In: *2011 IEEE international conference on computer vision (ICCV)*. Ieee. 2011, pp. 2548–2555.