AUTOMATIC CLASSIFICATION OF PHOTOGRAPHS AND GRAPHICS[*]

Yuanhao Chen[1], Zhiwei Li[2], Mingjing Li[2], Wei-Ying Ma[2]
[1]University of Science and Technology of China, Hefei 230026, China
[2]Microsoft Research Asia, 49 Zhichun Road, Beijing 100080, China

ABSTRACT

In general, digital images can be classified into photographs and computer graphics. This taxonomy is very useful in many applications, such as web image search. However, there are no effective methods to perform this classification automatically. In this paper, we manage to solve this problem from two aspects. At first, we propose some novel low-level features that can reveal perceptional differences between photographs and graphics. Then, we adopt an effective algorithm to perform the classification. The experiments conducted on a large-scale image database indicate the effectiveness of our algorithm.

1. INTRODUCTION

According to the ways in which they are generated, digital images can be classified into photographs and graphics. Photographs are often acquired by cameras and scanners, and graphics are generated by computers. The taxonomy is very useful in many applications, such as web image search, desktop search and image processing.

When searching for images on the web, we know both the semantic content and the type of images we want beforehand. For example, we may want to find cartoon pictures of dogs. A helpful step is to limit the search to graphics while filtering out the photographs of dogs. Unfortunately, current commercial image search engines, like Google and Yahoo Image Search, do not provide such functionalities. These search engines are only based on the textual information such as the surrounding text and the image filename. The textual information can describe the semantic content of images to a certain degree, but it can rarely distinguish image types. Therefore, the automatic classification of photographs and graphics can be used to improve the search experience by filtering out the images whose types are improper. Even when we do not have prior intensions, properly grouping images according to their types can help quickly locate the target images.

Another important application of the classification is desktop search. Personal photograph management is an important component of desktop search. The classification of photographs and graphics is needed as the first step of photograph management.

The classification also plays an important role in the optimization of image processing. Photographs and graphics have very different perceptional characteristics. Graphics look much simpler than photographs. If the characteristics are taken into account, the most appropriate method would be adopted to improve the performance of image processing.

Because of the many potential applications of the taxonomy, many methods have been reported for this problem [1, 4, 5].

In [1], Athitsos et al, used several features to measure the differences between photographs and graphics. The features used include the number of colors, most prevalent color, farthest neighbor metric, saturation metric, farthest neighbor histogram metric, and a few more. An error rate of 9% was reported for distinguishing images encoded by JPEG.

In [4], Lienhart and Hartmann proposed an algorithm to distinguish actual photos from computer-generation realistic-looking images such as ray tracing images or screen shots from photo-realistic computer games. They measured the amount of noise by means of histogram of the absolute difference image between the original and its de-noised version. Because the computer-generated images are less noisy than actual photos, this feature can distinguish between actual photos and computer-generated images.

Tian-Tsong Ng et al. solved the same problem of Lienhart and Hartmann in [5]. Motivated by physical image generation process, they used a geometry-based image model to tackle the problem.

Although many people have worked on this problem, the existing methods are not applicable to web images or large image collections. First, the computational cost of some algorithms is very high. The per-image feature-extraction time of [5] is more than 50 seconds. It is intolerable to web image search engines. Second, some features used before are not robust enough to noise. For example, images are usually resized in the web environment. Due to the interpolation used in the resizing process, the number of unique colors would greatly increase. So the performance of the feature using the number of colors would degrade significantly.

Based on these methods, we propose several new features such as the ranked histogram feature and the ranked

[*] The work was performed at Microsoft Research Asia.

1-4244-0367-7/06/$20.00 ©2006 IEEE

region size feature that can reveal the perceptional differences between photographs and graphics. These features exhibit promising performance with low computational cost.

In Web image retrieval, sometimes only the reliable results are needed. Therefore, we integrate a rejection option in the classification process. Ambiguous images such as mixed images have high probability to be rejected. And the classification accuracy is improved for the images not rejected.

The paper is organized in 7 sections. An overview of our classification method is presented in Section 2. The difference between photograph and graphics is analyzed in Section 3. Our proposed features as well as other traditional low-level features used for classification are illustrated in Section 4. The classifier is described in Section 5. The experimental result and discussions are given in Section 6. Finally, we conclude the paper in Section 7.

2. ALGORITHM OVERVIEW

We build a demonstration system to discriminate the photographs and graphics. Before the feature extraction, each image is normalized to a predefined size via nearest-neighbor interpolation. We perform this pre-processing for two reasons. One is the consideration of the computational cost. The time used in feature extraction is almost proportional to the image size. So reducing images to small size can greatly decrease the computational cost. The other reason is to facilitate the application in web image search engine. In a typical search engine, images are often stored in the thumbnail format due to copyright problem and disk space cost. Then some low-level features are extracted from each image for classification. Finally, the AdaBoosting learning algorithm proposed in [2] is used as our classifier to give the final judgment.

3. DIFFERENCE ANALYSIS

Photographs are the images acquired by digital cameras or scanners. And graphics are drawings and images generated by computer software. Examples of graphics are cartoons, slides, maps, charts and so on; examples of photographs are photos, scanned images and so on. However, there still are many ambiguous images such as photographs processed by some computer software. Due to the presence of such ambiguous images, we manually labeled all the images to decide their appropriate categories.

After observing a large amount of photographs and graphics, we find three major differences between the two classes:

- Graphics are often composite of several big single-color regions. However, the single-color regions rarely appear in photographs, because this kind of regions is not common in the real world and there is much noise in photo generation process.
- Photographs are often acquired by digital cameras. They depict the objects of the real world. Because of the texture of the objects and noise in the photo generation process, the texture information of photographs is very different from graphics.
- Certain colors are more likely to appear in graphics than in photographs, such as some highly saturated ones.

4. FEATURES USED FOR CLASSIFICATION

Based on the above observations of the differences between photographs and graphics, we proposed the following features for classification. Each of the features is carefully chosen to measure some aspects of the differences mentioned in Section 3.

4.1. The ranked histogram feature

The ranked histogram feature is used to replace most prevalence color feature proposed in [1]. As mentioned above, graphics tend to have fewer colors than photographs, and the percentage of the pixels of the prevalent colors for graphics is higher. So the percentage of pixels having the most prevalent color is used to distinguish the two classes. However, many graphics have more than one prevalent color. In such situation, the most prevalent color feature does not work well. As it is difficult to define a proper threshold that works well in all cases, we propose the ranked histogram feature to solve this problem.

First, for each image, we quantize the red, green, and blue channels into 32 bins respectively. A $32768=32^3$ bin histogram is created which give the count of each quantized color presented in the image. The histogram is normalized to unit length in L1 norm. Second, the histogram is sorted in descending order by the value of the elements. Finally, the *m* largest elements are retained as the ranked histogram feature.

In fact, many color quantization methods may be used for the extraction of ranked histogram feature. We choose this scheme for the sake of simplicity.

4.2. The color moment feature

Because certain colors occur more frequently in graphics than in photographs, the color distribution of graphics is very different from that of photographs. The color histogram is often used to measure the color distribution. However, the dimension of the color histogram feature is usually high. Therefore, the color moment feature [6] may be used instead. It also characterizes the color distribution, and produces more compact indices. We implement this feature as follows.

The first step is to transform the input image from the RGB color space to the HSV color space. Then the first three moments, mean, standard deviation and skewness, are computed for the H, S, and V channels respectively. The experiment shows that the color moment feature gives better performance on our dataset than the color histogram.

4.3. The correlograms feature

The correlograms feature is proposed by Jing et al. in [3]. It is considered to be one of the best features for content-based image retrieval. As it incorporates the spatial correlation of colors, it can provide the texture information to a certain degree. It could also be used to describe the global distribution of the local spatial correlation of colors. The information can effectively describe the differences in texture and color distribution between photographs and graphics.

In order to extract the feature efficiently, we simplify the process of the feature extraction. The first step is to quantize colors in the RGB color space to m color bins. Given any pixel of color c_i in the image, the probability that the neighbors of the given pixel are the identical color c_i is calculated as the feature.

4.4. The ranked region size feature

As indicated in Section 3, the single-color regions are rarely observed in photographs. In fact, there are many single-color regions in photographs. The sizes of these regions are very small and hard to discover. On the other hand, graphics are often composite of several large single-color regions. So the sizes of single-color regions could provide useful information for classification. There are many existing methods to segment an image into several regions. Our classification algorithm is insensitive to the image segmentation results, thus the accuracy of segmentation is not very important. So, we use a fast algorithm to implement the ranked region size feature.

First, we define the distance between two pixels. For two pixels p_1 and p_2, with colors (r_1, g_1, b_1) and (r_2, g_2, b_2), the distance between them is defined as $d = |r_1-r_2| + |g_1-g_2| + |b_1-b_2|$.

Second, a neighbor p_2 of p_1 is considered as in the same region of p_1 if the distance between them is smaller than a predefined threshold.

Finally, the watershed segmentation algorithm is used to segment the image into several regions. The sizes of the c largest regions are used as the ranked region size feature.

4.5. The farthest neighbor histogram feature

This feature is proposed by Athitsos et al. in [1] for classification of graphics and photographs. It can identify the patterns of the color transitions from pixel to pixel which is very different in graphics and photographs. Considering its powerful distinguishing ability, we use it as one of our features.

5. CLASSIFICATION ALGORITHM

Given a list of low-level features, we need a method to combine all these features into the final judgment. AdaBoosting learning algorithm is adopted as our classifier. It is a way to combine the performance of many "weak" classifiers to produce a powerful "committee". Weak classifiers can be very simple and only required to be better than the chance. The weak classifiers we used are single-split trees with only two leaf nodes. AdaBoosting with trees is thought to be one of best off-the-shelf classifiers so far. It works effectively with high dimensional data and the computational cost is very low in the judgment phase. The two advantages fit the need of our system very well.

6. EXPERIMENTAL RESULT

We have evaluated the performance of our system on a large image database consisting of about 36,000 graphics and 35,000 photographs, which are obtained from two sources. About 35,000 images are downloaded from special web sites such as the photography forums and the web sites which offer clipart images and cartoons. In this way, we can guarantee the images vary in content. Other images are downloaded from an image search engine. First we select many queries such as map, chart and the name of famous stars and submit these queries to the PicSearch image search engine (www.picsearch.com). The first 500 returned images are downloaded and labeled by hand. Before extracting the low-level features, all images are resized to 160 pixels on the larger dimension, and proportionally on the other except those images whose pixels on the larger dimension are less than 160.

In our experiment, five-fold cross validation is used to validate our algorithm. More precisely, we randomly divide all images into five equal-size sets. In each run of cross-validation, we pick one set as the testing set, the other sets as the training set.

First we give the classification result of each individual feature in Table1. E_g is the error rate for graphics in testing set, while E_p is the error rate for photographs. E is the average error rate. In the table, we see that the features we proposed exhibit better performance than the existing features in discriminating photographs and graphics. The ranked histogram feature outperforms the most prevalent color feature by 2.3% in precision. And the error rate of the color moment feature is lower than that of color histogram.

The performance of our system is presented in Table 2. We compare our system with the algorithm proposed by Athitsos et al. in [1]. The 9 types of features used by them are: color histogram, fastest neighbor histogram, saturation, prevalent color, number of colors, and dimension ratio and a few more. And 5 types of features presented in Section 4

Table 1: Classification performance of each feature

Features	E_g	E_p	E
Color histogram	15.4	12.2	13.8
Color moment	14.9	10.4	12.7
Ranked histogram	19.1	14.5	16.8
Most prevalent color	19.9	18.1	19.1
Ranked region size	19.1	17.9	18.5
Correlogram	13.7	10.9	12.3
Fastest neighbor histogram	13.9	11.4	12.7

Table 2: Comparison with Swain's algorithm

	E_g	E_p	E
Athitsos's algorithm	7.6	9.5	8.6
Our algorithm	6.8	4.3	5.5

are used in our algorithm. Both Athitso's algorithm and our algorithm use the AdaBoosting classifier for impartiality. In Table 2, we see that the error rate of Athitsos's algorithm is 8.6%, which is very similar to 9% they presented in their paper. And the error rate of our algorithm is 5.5%, which is 36% lower than theirs.

The examples of the misclassified images are showed in Figure 1. The misclassified photographs are photos either made up of only a few colors or de-noised by some image processing tools. The graphics misclassified are mostly colorful cartoons and painting.

In image retrieval, reliable search results are always needed. Setting a rejection threshold can help us get those images in which the classifier has a high confidence. The precision with rejection rate is showed in Figure 2. In the figure, we can see that the error rate is greatly decreased with low rejection rate.

We measured the computational cost of our algorithm. Our experiment is conducted on a computer with Intel P4-Xeon 3.1GHz CPU, 3G memory. The per-image feature-extraction time is about 5ms. It means that one CPU can process about 200 images per second, which is acceptable for web search engine.

7. CONCLUSION

We present an effective and efficient algorithm for classifying photographs and graphics based on image content. First, we analyzed the perceptual differences between the two classes. Then we proposed several novel features such as the ranked histogram feature and the ranked region size feature to measure the differences. Finally, the AdaBoosting Algorithm is used as our classifier to give the final result. When the reliable results are needed, the rejection rate can be used to get the images with high confidence. Using a large image dataset, our algorithm shows promising performance.

Our work is not a full solution for classification. Much useful information is not used in our algorithm. For example, the photographs acquired from cameras often contain some metadata such as the focus length. It would be beneficial for classification. And the surrounding text is also helpful. We will combine these kinds of information in our algorithm in the future work.

8. REFERENCES

[1] V. Athitsos, M. J. Swain, and C. Frankel, "Distinguishing Photographs and Graphics on the World Wide Web", *in IEEE Workshop on Content-Based Access of Image and Video Libraries*, pp. 10-17, June 1997.

[2] J. Friedman, T. Hastie, and R. Tibshirani, "Additive Logistic Regression: A Statistical View of Boosting", *Technical Report*, Stanford University, 1998.

[3] J. Huang, R. Kumar, and M. Mitra, "Image Indexing Using Color Correlograms", *Proc. CVPR*, pp. 762-768, 1997.

[4] R. Lienhart and A. Hartmann, "Classifying Images on the Web Automatically", *Journal of Electronic Imaging*, 11(4), pp. 445-454, 2002.

[5] T. T. Ng, S. F. Chang, and J. Hsu, "Physics-Motivated Features for Distinguishing Photographic Images and Computer Graphics", *in proceedings of ACM conference on Multimedia*, pp. 239-248, Singapore,2005.

[6] M. Stricker, and M. Orengo, "Similarity of color images", *in Proceedings of SPIE Storage and Retrieval for image and Video Databases Conference*, pp. 381-392, 1995.

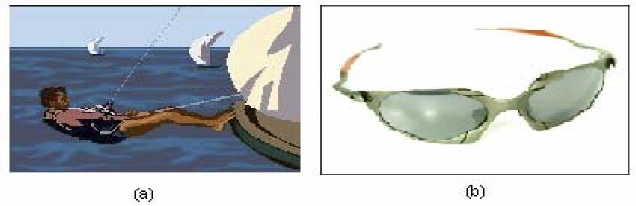

Figure 1: Examples of misclassified images

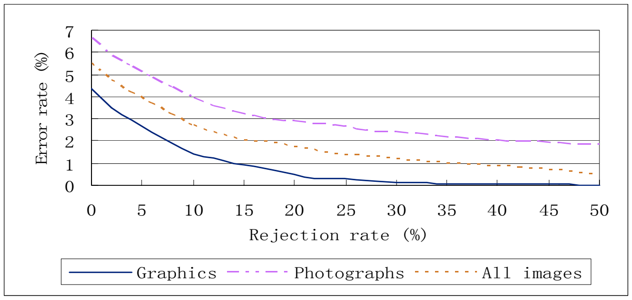

Figure 2: Error rate vs. rejection rate