ICIC Express Letters
Part B: Applications
Volume **7**, Number **4**, April **2016**

ICIC International ©2016 ISSN 2185-2766
pp. **797–803**

HIERARCHICAL FAST IMAGE INDEXING USING GLOBAL AND LOCAL FEATURES

Chengshun Yang[1], Le Ge[1] and Zhong Yang[2]

[1]School of Electric Power Engineering
Nanjing Institute of Technology
No. 1, Hongjing Avenue, Nanjing 211167, P. R. China
chengshunyangnit@sina.com

[2]College of Automation Engineering
Nanjing University of Aeronautics and Astronautics
No. 29, Yudao Street, Nanjing 210016, P. R. China

Received October 2015; accepted January 2016

Abstract. *In this paper we study the content-based image searching with a fast ranking algorithm. First, the global image features based on sparse histogram are analysed. The similarity between different image content is computed. Second, the local image features based on scale-invariant feature transform and speeded up robust features are analysed. Based on the detailed discrimination among images we are able to achieve a reliable ranking of similar images. Third, a hierarchical image indexing framework is proposed, in which the global features are used to eliminate the target images with mismatching colors and the local features are used to detect the differences in detailed image structures. Experimental results show that the proposed method is very efficient compared with the methods using only texture features.*
Keywords: Image indexing, Local features, Global features, Fast algorithm

1. **Introduction.** Image content-based searching is a very valuable yet challenge task in computer vision [1, 2]. Past researches in content-based image understanding can be classified into two groups: the global statistics based methods and the local feature based methods. The first group of methods makes use of the global statistics [3] on an image, such as color histogram [4], and color distributions [5]. These methods can model the global changes but they cannot reflect the details in content. The second group of methods is based on the local features [6] to model the detailed structure of an image, such as points of interest [7], edges [8] and texture features [9]. These methods usually require a lot of computation time.

Torralba et al. [3], studied the importance of global features for human visual perception, and their results suggested that the global features might be very important for fast image scene recognition. Lin et al. [4], proposed a fast algorithm using histogram features of images. In their study, efficient histogram features were adopted and the accuracy could be further improved. Zhou and Huang [8], studied edge information in image content understanding. The results showed that edge based systems could effectively model the image structures. Vipparthi and Nagar [10], adopted multiple histogram features to jointly model the image content for indexing. However, the discriminant ability in similar images needs to be improved. In the past works, the global features and the local features are rarely used together. The color features may bring fast computing speed and relatively low accuracies. The texture features, on the other hand, can ensure high image matching accuracies, but increase the computational costs.

In this paper, we proposed a two-stage hierarchical image indexing method. In the first stage, images with large distances in global color features are removed from the target dataset. In the second stage, detailed local features are used for accurate object matching in a small amount of images selected in the first stage.

The rest of the paper is organized as follows: Section 2 provides the histogram based features for fast indexing; Section 3 provides the local feature analysis method; Section 4 gives the description of the proposed two-stage hierarchical searching algorithm; experimental results are given in Section 5; finally, the conclusions are drawn in Section 6.

2. Histogram Feature Analysis.

Histogram feature is a widely used global statistic feature that reflects the color percentage in an image.

In this section, we will give the detailed description of how to extract the histogram based global features adaptively according to different probe image. We suppose there is at least one image in the gallery (our image database) that contains the probe image or a part of the probe image. In practice, when we search for a particular image in the database, it may be confused by various backgrounds or partially occluded. Therefore, we need to design a robust histogram based feature that is not dependent on the various backgrounds in an gallery image. Furthermore, the desired features should be able to handle a reasonable level of occlusions.

Given a probe image, the histogram is the statistics over all types of RGB color. Before we extract the histogram, we need to compress the color space for the sake of computational efficiency. The color of a pixel $I_{x,y}$, where x and y are the coordinates of the 2D image, is represented by a three-dimensional vector as shown in Equation (1).

$$I_{x,y} = [R_{x,y}, G_{x,y}, B_{x,y}] \tag{1}$$

where each color component takes discrete values: $R_{x,y}=0, 1, \ldots, 255$, $G_{x,y}=0, 1, \ldots, 255$, $B_{x,y} = 0, 1, \ldots, 255$. The color space compression we referred to in this section directly quantizes the color component in to a smaller scale as shown in Algorithm 1.

Algorithm 1 Color space compression and histogram feature construction

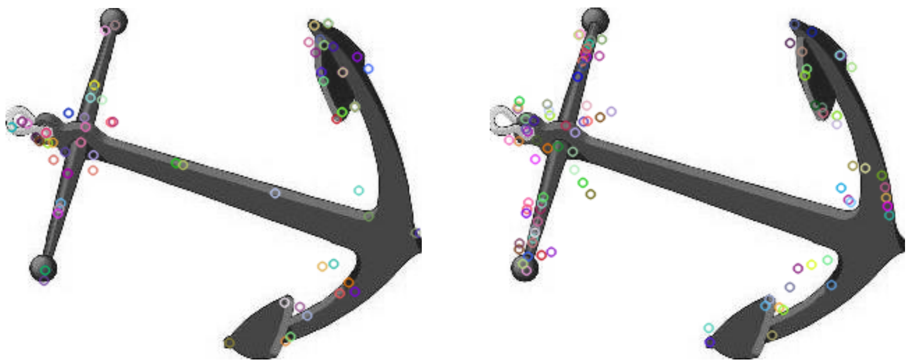
Require:
  Probe image pixels $I_{x,y}$
Ensure:
  Histogram feature from compressed color space H_c

1: **for** $x = 0$ to *height* **do**
2:   **for** $y = 0$ to *width* **do**
3:     Read pixel $I_{x,y} = [R_{x,y}, G_{x,y}, B_{x,y}]$;
4:     Compute color index $c = R \times (256/D)^2 + G \times (256/D) + B$, where D is the compress ratio;
5:     Accumulate histogram $H_c + +$.
6:   **end for**
7: **end for**

The global color histogram based features for each image in the gallery database are computed according to a set of selected colors in the histogram of the probe image. As shown in Equation (2), the selected histogram value $H_c^{gallery}$ of the gallery image forms a set of global features $\mathcal{F}$.

$$\mathcal{F} = \left\{ f_c | f_c = H_c^{gallery},\ c \in \phi \right\} \tag{2}$$

where ϕ is the set of compressed RGB color indexes whose histogram value H_c in the probe image is above a certain threshold th. c is the index of color. Since the histogram features are mapped from the probe image, they are not dependent on the backgrounds in the gallery images.



(a) SIFT (b) SURF

FIGURE 1. Extracted SIFT and SURF features on an example image: (a) SIFT features; (b) SURF features

3. **Local Feature Analysis.** SIFT (Scale-Invariant Feature Transform) [11] is a powerful two-dimensional image feature that is widely used in computer vision applications such as object recognition, navigation and video tracking. SIFT can robustly match images under partial occlusion, and the feature descriptor is invariant to uniform scaling, orientation. It also tolerates affine distortion and illumination changes.

Another two-dimensional image feature we adopted in this paper is SURF (Speeded Up Robust Features) [12]. As the name suggested, it is generally faster than SIFT. It is widely used in object recognition, image matching and registration. The standard version of SURF is several times faster than SIFT and it is more robust against different image transformations than SIFT. As shown in Figure 1, the extracted features on an example image using SIFT features and SURF features are given.

Suppose the feature descriptor based on the local feature (either SIFT or SURF) is denoted as $S_{i,j}$, where i denotes the index of an image from the database and j denotes the index of the feature descriptor of that image. The process to find the closest features between two images is shown in Algorithm 2. $\mathcal{M}$ is a set of matched image pairs. L is the total number of images to match. Generally speaking the computational cost increases significantly when L is a large number. We are able to reduce the number of potential match candidate images by the ranking based on the global feature. The process is described in Section 4.

Algorithm 2 Image matching using closest local feature

Require:
Image feature descriptors $S_{i,j}$
Ensure:
Matched pairs of images $\mathcal{M}$

for $i = 0$ to L, where L is the total number of images in the database **do**
    for $i' = i$ to L **do**
        Calculate distance between two features descriptors $d = |S_{i,j} - S_{i',j'}|$;
        Find the minimum value of the above distance: $\forall j, j'$, $d^i_{\min} \le d$;
        if $d^i_{\min} < d^{i-1}_{\min}$ **then**
            Update the closest pair for the current i^{th} image: $\mathcal{M}^i = \mathcal{M}^{i-1} \bigcup (i, i')$.
        end if
    end for
end for

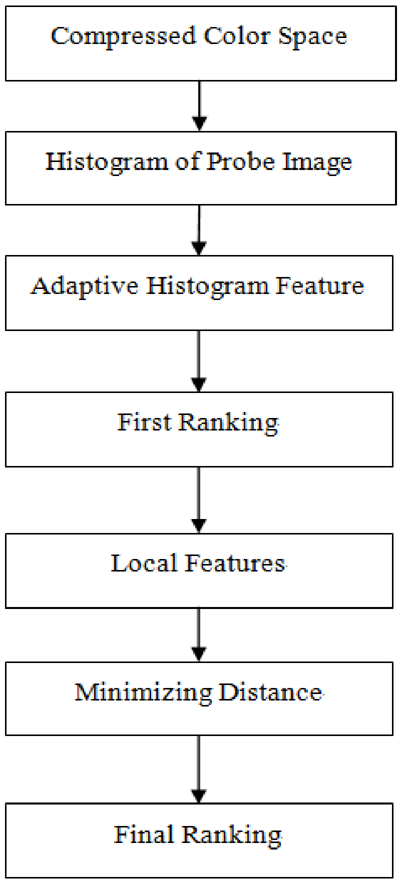


FIGURE 2. Flowchart of the two-stage ranking algorithm

4. **Hierarchical Fast Image Indexing.** In this paper, we propose a two-stage hierarchical searching algorithm, as shown in Figure 2, to speed up the feature matching process described in Algorithm 2. In the first stage, an initial ranking is done based on the histogram based features as shown in Equation (3).

$$R^{(1)} = sort\left(\sum_{c} f_c^0 - f_c^l\right) \tag{3}$$

where f_c^0 denotes the feature of probe image, f_c^l denotes l^{th} image in the gallery. Function $sort()$ returns a sequence of values sorted from smallest to largest.

Since the color histogram feature is based on the global characters of the image color distribution, it is able to eliminate images with obvious color difference from the target image. The computational cost of histogram is low after RGB color compression. The computational cost of the feature generation in Equation (2) is even lower when we simply move the pointers in the c implementation.

In the second stage, we select the first n images in the ranking $R^{(1)}$ to perform image matching based on local features (either SIFT or SURF) as described in Section 3. The final ranking $R^{(2)}$ is achieved with reduced complexity from $L \times L$ to $n \times n$. When the ranking $R^{(1)}$ is reliable, we generally set n to a small number, such as 5. The balance between accuracy and speech can be adjusted by ratio n/L according to specific applications.

5. **Experimental Result.** We use Caltech 101 database [13] for image indexing experiment. There are about 40 to 800 images per category and most categories have about 50 images. Examples used in this experiment are shown in Figure 3.

FIGURE 3. A depiction of example images from Caltech 101 database

TABLE 1. Probability of ranking accuracy using various algorithms

Algorithms	**Prob(1)**	**Prob(2)**	**Prob(3)**
Hist Only	67.1%	76.3%	83.9%
SIFT Only	97.3%	98.7%	99.3%
SURF Only	96.5%	97.2%	98.1%
Two-Stage (SIFT)	96.9%	98.1%	99.1%
Two-Stage (SURF)	96.0%	96.9%	98.1%

The testing results are shown in Table 1. We use three types of definitions to evaluate the accuracy of the ranking result. "Prob(1)" stands for the probability of target image ranked at the first place in the final ranking. "Prob(2)" stands for the probability of target image ranked within the first two places in the final ranking. "Prob(3)" stands for the probability of target image ranked within the first three places in the final ranking. We can see that when the target image is ranked at the first place, it is considered as a successful indexing. When the target image is ranked within the first three places, we may still recommend the first three images to the user as the searching result. In the algorithms column "Hist Only" stands for the first stage of the proposed ranking algorithm that uses only the global histogram feature. "SIFT Only" stands for the traditional image matching algorithm that uses local features extracted by SIFT algorithm. Similarly, "SURF Only" stands for using SURF algorithm to extract local features. "Two-Stage (SIFT)" stands for the proposed two-stage hierarchical ranking algorithm with SIFT features. Similarly, "Two-Stage (SURF)" stands for using SURF features. According to Table 1, the SIFT features bring slightly better results than SURF features. "Hist Only" algorithm gives the lowest accuracy, as much of the detailed structure information is not considered in order to improve the computing speed. We still need to evaluate the speed of the algorithms using different features to draw the final conclusion.

In Figure 4, we can see that speeds of the different algorithms are very different. The speed is calculated under different image sizes. As the image size goes up, the proposed algorithm still maintains a high speed. The traditional SIFT and SURF based image matching algorithms are influenced by the image sizes. Notice that we may resize the image to speed up feature extraction process, but the resize process may also decrease the accuracy of feature matching. Although "SIFT Only" algorithm gives the highest accuracy, it also requires the longest computing time. All the images in the databases are

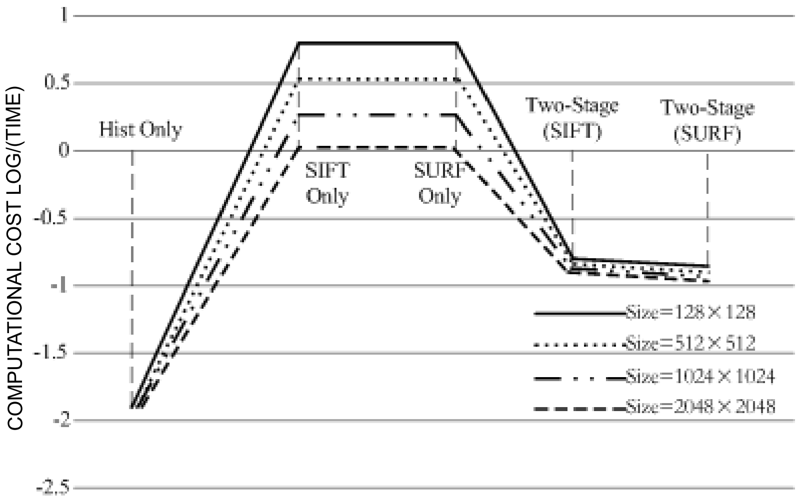


Figure 4. Comparison of computational time of different algorithms

included in the matching process using SIFT features. On the other hand, the proposed "Two-Stage (SURF)" algorithm is very fast, and it is able to maintain a high accuracy at the same time. Only a few images from the database are selected for matching by the first stage ranking based on "Hist Only" algorithm. In practice, the proposed ranking algorithm has a better balance between accuracy and speed, which are the two most important factors in image indexing.

6. **Conclusions.** In this paper a hierarchical searching framework is proposed. We analysed the challenges in content-based image searching, and the existing matching algorithms based on SIFT or SURF features are used for comparison. In order to speed up the image searching process, we introduced global features based on color histogram. In the first layer of the proposed framework, the target image was used to generate sparse histogram features and we used it to match the large amount of images in the searching database. In the second layer of the framework we used local features to further discriminate the detailed structure differences in images to improve the ranking accuracy. By combining both global color features and local texture features we were able to reduce the computational cost and maintain the ranking reliability. In future work, we will further explore the possibility of using semantic meaning to improve image searching.

Acknowledgement. This work was supported by High-level Scientific Research Foundation for the introduction of talent of Nanjing Institute of Technology (YKJ201412).

REFERENCES

[1] A. Goodrum and A. Spink, Image searching on the Excite Web search engine, *Information Processing & Management*, vol.37, no.2, pp.295-311, 2001.
[2] P. C. Yuen and C. H. Man, Human face image searching system using sketches, *IEEE Trans. Systems, Man and Cybernetics, Part A: Systems and Humans*, vol.37, no.4, pp.493-504, 2007.
[3] A. Torralba, A. Oliva, M. S. Castelhano and J. M. Henderson, Contextual guidance of eye movements and attention in real-world scenes: The role of global features in object search, *Psychological Review*, vol.113, no.4, pp.766-786, 2006.
[4] C. H. Lin, C. C. Chen, H. L. Lee and J. R. Liao, Fast K-means algorithm based on a level histogram for image retrieval, *Expert Systems with Applications*, vol.41, no.7, pp.3276-3283, 2014.
[5] J. Sun, X. Zhang, J. Cui and L. Zhou, Image retrieval based on color distribution entropy, *Pattern Recognition Letters*, vol.27, no.10, pp.1122-1126, 2006.
[6] S. Jothimani and P. Betty, Image authentication using global and local features, *International Conference on Green Computing Communication and Electrical Engineering*, Coimbatore, pp.1-5, 2014.
[7] Y. Xia, S. Luo, X. Zhang, and H. Y. Bae, A novel retrieval method for multimodal point of interest data, *International Journal of Multimedia and Ubiquitous Engineering*, vol.9, no.7, pp.281-296, 2014.

[8] X. Zhou and T. Huang, Edge-based structural features for content-based image retrieval, *Pattern Recognition Letters*, vol.22, no.5, pp.457-468, 2001.

[9] X. Wang, B. Zhang and H. Yang, Content-based image retrieval by integrating color and texture features, *Multimedia Tools and Applications*, vol.68, no.3, pp.545-569, 2014.

[10] S. K. Vipparthi and S. K. Nagar, Multi-joint histogram based modelling for image indexing and retrieval, *Computers & Electrical Engineering*, vol.40, no.8, pp.163-173, 2014.

[11] D. G. Lowe, Distinctive image features from scale-invariant keypoints, *International Journal of Computer Vision*, vol.60, no.2, pp.91-110, 2004.

[12] H. Bay, T. Tuytelaars and L. van Gool, Speeded-up robust features (SURF), *Computer Vision and Image Understanding*, vol.110, no.3, pp.346-359, 2008.

[13] L. Fei-Fei, R. Fergus and P. Perona, Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories, *IEEE CVPR Workshop on Generative-Model Based Vision*, 2004.