Robust Image Feature Point Matching Based on Structural Distance

Maodi Hu[1(✉)], Yu Liu[2], and Yiqiang Fan[1]

[1] Digital Technology Academy, Aisino Corporation, Beijing 100195, China
Londeehu@gmail.com
[2] Center of Information and Network Technology,
Beijing Normal University, Beijing 100875, China

Abstract. Feature point matching is a key step of image registration, object recognition and many other computer vision applications. By using the proposed structural distance between feature point sets as the matching similarity, we are able to match the spatial structures of feature points in different images. In the optimization process of the structural distance, both local and global relationship are considered, which greatly improves the robustness and accuracy. We also present a fast algorithm with higher efficiency, which approximately realizes this method by linear matrix multiplication operations. The proposed method achieve promising matching results in the experiments.

Keywords: Structural distance · Feature point matching · Iterative algorithm

1 Introduction

Representation and matching of images are the core technologies of many computer vision and pattern recognition applications [1][2]. Extraction of feature points with certain modes is commonly used for image representation. Therefore, feature point matching between different sets becomes an important part of vision tasks, such as image retrieval, 3D reconstruction, and panoramas. Due to the fundamental position of image feature point matching, its accuracy may severely influence the final output. Normally, the main goal of the existing feature point matching methods is measuring the distance between feature point descriptors extracted from different images.

SIFT (scale invariant feature transform) [3][4] and SURF (speeded up robust features) [5][6] are the two most commonly used image feature point extraction methods. The main steps of SIFT include constructing scale spaces, detecting Gauss differential extreme points, filtering feature points by the second order Taylor expansion and the Hessian matrix, assigning gradient direction value to generate descriptor. SURF can be regarded as an accelerated algorithm of the SIFT, which uses the Hessian matrix of the integral map for feature point detection, and uses responses of Haar wavelet to build descriptors. Its calculation speed is better than SIFT. Generally, image feature point extraction module is divided into the detection part and the description part, which can be chose according to the application requirements.

T. Tan et al. (Eds.): IGTA 2015, CCIS 525, pp. 142–149, 2015.
DOI: 10.1007/978-3-662-47791-5_17

Traditional image feature point matching methods did not consider the location relationship of feature points, which easily leads to many errors, for example, feature points on an object densely match to multiple other objects. Inspired by the PageRank [7] algorithm, the proposed method employs the weighted voting strategy, which iteratively approximates a translation-invariant and rotation-invariant structural distance between feature points from different images.

2 Definition of Structural Distance

The proposed method in this paper is mainly aimed at the problem of matching feature points of two images. Hereinafter we call the two images as image 1 and image 2. Note that, multiple image matching can be extended similarly. The feature points of two images are extracted using SIFT method, whose detection and description steps are introduced in [3][4]. Afterwards, SIFT feature points of the two images are then matched using the following algorithm.

Before the illustration of the matching process, we make the following definition and initialization. Let N and M respectively represent the number feature points extracted from image 1 and image 2. For the i-th point of image 1, let s_i and p_i denote its descriptor and location respectively, where i=1 ... N. Similarly, for the j-th point of image 2, let t_j and q_j denote its descriptor and location respectively, where j=1 ... M. Let D (matrix of N * M) be the distance matrix of feature point descriptors extracted from the two images, where D_{ij} is the Euclidean distance between the i-th feature point descriptor of image 1 and j-th feature point descriptor of image 2, which is calculated as $D_{ij} = \| s_i - t_j \|$. Let $\widetilde{W}$ (matrix N * N) be the proximity matrix of feature point locations in image 1, whose elements are

$$\widetilde{W}_{ik} = \begin{cases} e^{-\alpha \| pi - pk \|} & i \neq k \\ 0 & i = k \end{cases},$$

where α controls the influence of the spatial distance on the weight, k=1 ... N. The weights are then normalized as

$$W_{ik} = \frac{\widetilde{W}_{ik}}{\sum_{k=1}^{N} \widetilde{W}_{ik}}.$$

W is weight matrix of the structural distance for image 1. In other words, for the i-th feature point in image 1, W_{ik} is the weight from the k-th feature point s_k to the i-th feature point s_i. Note that, the sum of weights from all the feature point in image 1 to the i-th feature point is 1. Similarly, let $\widetilde{W}'$ (matrix N * N) be the proximity matrix of feature points in image 1, whose elements are

$$\tilde{W}'_{jl} = \begin{cases} e^{-\alpha \| pj - pl \|} & j \neq l \\ 0 & j = l \end{cases},$$

where l=1 ... M. The weights are then normalized as

$$W'_{jl} = \frac{\tilde{W}'_{jl}}{\sum_{l=1}^{M} \tilde{W}'_{jl}}.$$

W' is weight matrix of the structural distance for image 2. Let C (matrix of N * M) be the structural distance matrix of feature points extracted from the two images, whose initial value is C=D. C_{ij} represents the structural distance between the i-th feature point of image 1 and j-th feature point of image 2.

3 Feature Point Matching

In this work, the feature point matching problem is solved by optimizing structural distance for each feature point. For any feature point in one image, its optimal match in another image is the feature point with minimal structural distance between them. We use a voting strategy to optimize the structural distances between two feature point sets. The main idea is to pass the structure of distance information of each feature point to other points, according to their spatial location relations and descriptor distances.

In the classical PageRank algorithm [7], a directed graph is constructed to represent pages through links, the PageRank value is the sent out as weights for out links. After several rounds of iterations, the final PageRank value of each page can be converged. In each round of PageRank value updating, each page accepts updating weights from other pages to obtain a new PageRank value. The PageRank algorithm uses a graph-based voting strategy, where each vertex in a graph spread its value through edges gradually until convergence.

Image feature point matching is more complex, since it has to consider the spatial relations, the similarities between feature point descriptors, and other problems. The main idea of the proposed algorithm is as follows. For each feature point in an image, we consider a sub-graph structure containing the point, and optimize the distance to every sub-graph in another image in each iteration, where the change of structural distance of one feature point can affect the structural distances of all the other feature points, especially the neighboring ones.

For any i-th feature point in image 1 and any j-th feature point in image 2, the structural distance between the two feature points is a sum of three items. The first item is a Euclidean distance of these two feature point descriptors. The second item is the weighted sum of optimal structural distance of all the other feature points in image 1, where the optimal structural distance of each feature point in image 1 is the distance to its optimal matching point in image 2. Similarly, the third term is the weighted sum of optimal structural distance of all the other feature points in image 2, where the optimal structural distance of each feature point in image 2 is the distance to its optimal matching point in image 1. In each iteration, the structural distance between every feature point in image 1 and every feature point in image 2 is updated. The updating process is as follows. For all i and j, i.e., i = 1 ... N, j = 1 ... M, we updates

$$C_{i,j} = D_{i,j} + \sum_{k=1}^{N} W_{i,k} T_k + \sum_{l=1}^{M} W'_{j,l} T'_l \text{ ,}$$

where T is the optimal structural distance vector of image 1, T_k is the optimal structural distance vector of the k-th feature point. For k = 1 ... N,

$$T_k = \min_j C_{k,j} \text{ .}$$

Similarly, T' is the optimal structural distance vector of image 1, T'_l is the optimal structural distance vector of the k-th feature point. For l = 1 ... M,

$$T'_l = \min_i C_{i,l} \text{ .}$$

The optimization is finished until there is no element of C changes in one iteration, or the number of iterations reaches the preset maximum number.

4 Fast Algorithm

The algorithm updates T and T' every time after updating C. Considering these changes of T and T' are mostly trivial before and after each update of C, they can be updated in one time after all elements of C are updated. This approximation has few effect on the resulted values of elements of C. In conclusion, the robust feature point matching method based on structure distance can be simplified as the following fast algorithm by matrix operations.

```
Step 1:
  Initialize D, W, W', C as follows. For i=1 ... N,
j=1 ... M,
```

$$D_{ij} = \| s_i - t_j \| \text{ .}$$

```
For i=1 ... N, k=1 ... N,
```

$$\tilde{W}_{ik} = \begin{cases} e^{-\alpha \| pi - pk \|} & i \neq k \\ 0 & i = k \end{cases} \text{ .}$$

For i=1 ... N,

$$W_{ik} = \frac{\tilde{W}_{ik}}{\sum_{k=1}^{N} \tilde{W}_{ik}}.$$

For j=1 ... M, l=1 ... M,

$$\tilde{W}'_{jl} = \begin{cases} e^{-\alpha \| pj - pl \|} & j \neq l \\ 0 & j = l \end{cases}.$$

For j=1 ... M,

$$W'_{jl} = \frac{\tilde{W}'_{jl}}{\sum_{l=1}^{M} \tilde{W}'_{jl}}.$$

C=D.

Step 2:
Calculate

$$T = \min^{2} C,$$

$$T' = \min^{1} C,$$

where $\min^2$ is the minimum values of all the columns, and $\min^1$ is the minimum values of all the rows, and calculate

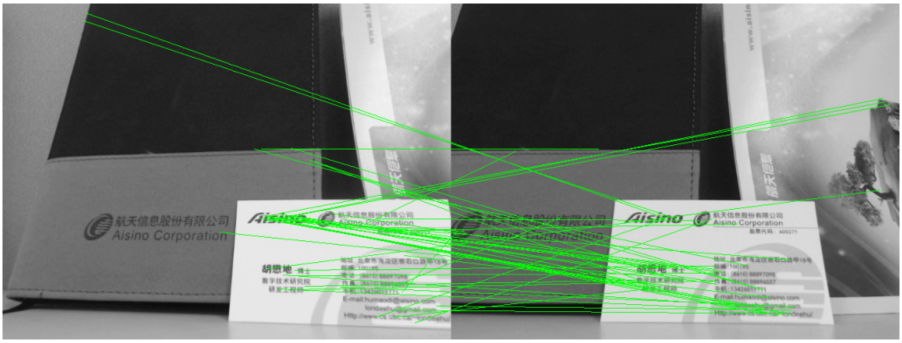
$$C = D + WT + W'T'.$$

Step 3:
If there is no element in C changes before and after Step 2, or the number of iterations reaches the preset maximum number, the optimization is over;
else, run Step 2 again.

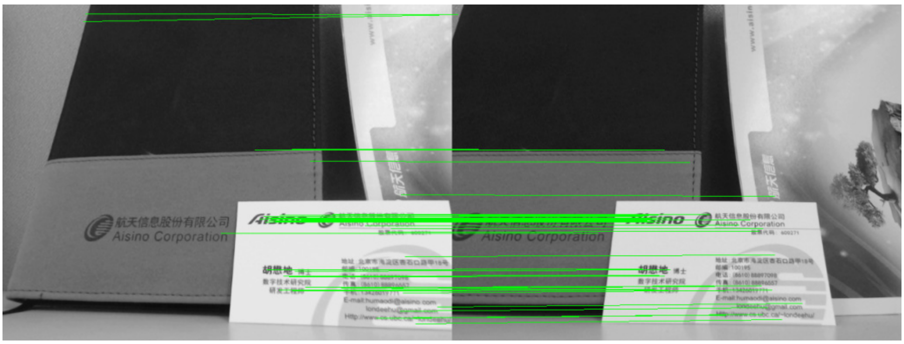
5 Experimental Results

We collect a large number of experimental images, and use qualitative and quantitative analysis to verify the correctness of the proposed method. The following experiment is based on the implementations of the proposed fast algorithm using Python + Opencv + Numpy + Scipy. The experimental image resolution is 640 x 480, and the preset maximum number of iterations to 10. In a 32-bit operating system running on a machine with 2.9 G dual-core CPU and 3.34 G memory, average process time of feature point matching is 0.03 seconds, which satisfies the requirement of real-time video analysis.

Figure 1 and Figure 2 are the comparison of two feature point matching methods, where (a) and (b) show the matching results of the nearest neighbor method and the proposed method respectively. The experimental process is as follows. Firstly, we place the target objects, rotate the camera to obtain two images of different viewing angle. Secondly, we extract feature points of the two images using the SIFT method. Finally, we match SIFT feature points by the nearest neighbor method and the proposed method. After the calculation of the Euclidean distances between feature point descriptors, (a) is obtained by the nearest neighbor matching, (b) is obtained by the proposed method. The green lines in (a) and (b) show the top 50 optimal matches calculated by these two methods. We can see that, for the feature points on notepad, business card, telephone and other objects, the proposed method successfully find most matching points, and is clearly more robust than the commonly used nearest neighbor method.

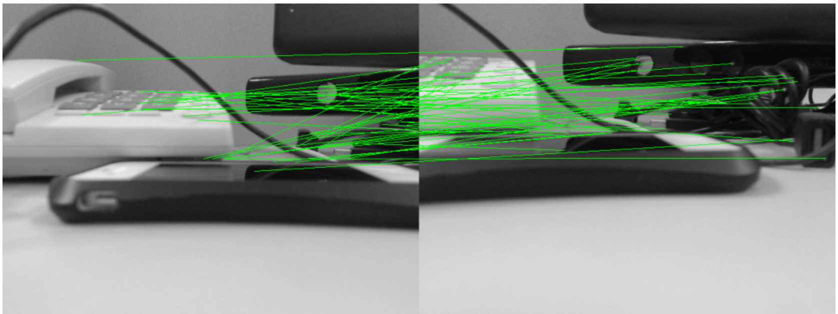


(a). Matching results of the nearest neighbor method

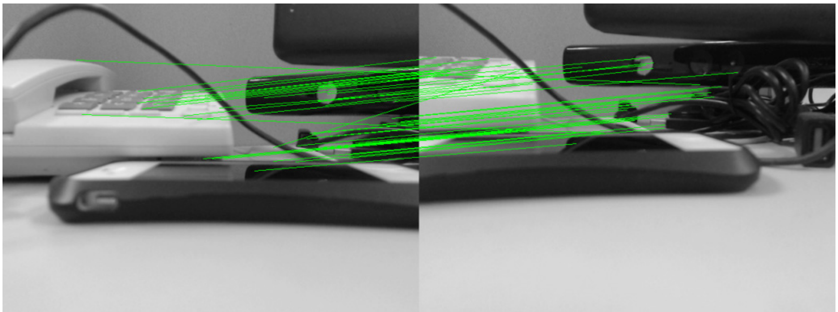


(b). Matching results of the proposed method

Fig. 1. Comparison of two matching methods



(a). Matching results of the nearest neighbor method



(b). Matching results of the proposed method

Fig. 2. Comparison of matching results

In addition, we use SIFT feature points together with the nearest neighbor method and the proposed method for object recognition experiments. Figure 3 is the results of seven categories of common objects, such as mobile phone, print, glass. In the process of experimental image collection, factor changes including viewing angle, object distance, and light source are considered. For correct recognition rate of all the objects, the proposed methods is far superior to the nearest neighbor method. Especially for objects with obvious texture structure such as print, display, and keyboard, the proposed method largely increase the correct recognition rate.

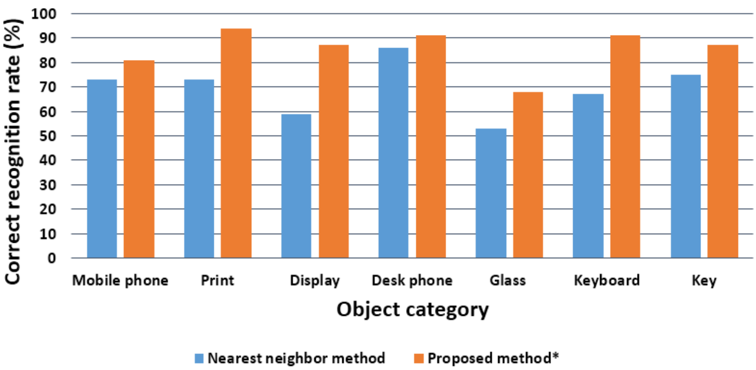


Fig. 3. Comparison of correct recognition rates

6 Conclusions

This paper proposes a feature point matching method based on structural distance. The proposed method gains better performance in robustness, by incorporating the graph-based structure information in feature points. Experiments show that the proposed matching method performs better than the commonly used nearest neighbor method. Moreover, this method is of high efficiency and can meet the requirement of the real-time video analysis.

Acknowledgement. This work is supported by industry research special funds for public welfare project No.201313008.

References

1. Maodi, H., Wang, Y., Zhang, Z., Little, J.J., Huang, D.: View-invariant Discriminative Projection for Multi-view Gait-based Human Identification. IEEE Transactions on Information Forensics and Security **8**(12), 2034–2045 (2013)
2. Maodi, H., Wang, Y., Zhang, Z., Zhang, D., Little, J.J.: Incremental Learning for Video-based Gait Recognition with LBP-Flow. IEEE Transactions on Cybernetics **43**(1), 77–89 (2013)
3. Lowe, D.G.: Object recognition from local scale-invariant features.In: Proceedings of the International Conference on Computer Vision, pp. 1150–1157 (1999)
4. Lowe, D.G.: Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision **60**(2), 91–110 (2004)
5. Bay, H., Tuytelaars, T., Van Gool, L.: SURF: speeded up robust features. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006, Part I. LNCS, vol. 3951, pp. 404–417. Springer, Heidelberg (2006)
6. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: SURF: Speeded Up Robust Features. Computer Vision and Image Understanding **110**(3), 346–359 (2008)
7. Page, L., Brin, S., Motwani, R., Winograd, T.: The PageRank citation ranking: Bringing order to the Web. Technical report SIDL-WP-1999-0120, Stanford InfoLab (1998)