A Tree-Structured Feature Matching Algorithm

Xiongwei Sun[1], Xiubo Ma[2], Lei Chen[1], Li Wan[1], and Xinhua Zeng[1](✉)

[1] Hefei Institutes of Physical Science, Chinese Academy of Science, Hefei, China
xiongweisun@163.com, xhzeng@iim.ac.cn
[2] Department of Computer Engineering, AnHui SanLian University, Hefei, China
maxiubo0310@126.com

Abstract. Feature matching is essential in computer vision. In this paper, we propose a robust and reliable image feature matching algorithm. It constructs several matching trees in which nodes correspond to traditional sparsely or densely sampled feature points, and feature lines are constructed between the nodes to build a cross-references based on a Difference-of-Gaussians down-sampling pyramid. This can make patch-based descriptors combine efficiently with spatial distributions. By comparing with SIFT, SURF and ORB, our method can get much more correct correspondences on both synthetic and real data under the influence of complex environments or transformations especially in irregular deformation and repeated patterns.

Keywords: DoG · Image feature matching · Tree structured matching · Feature line

1 Introduction

Feature correspondence is a fundamental task in many applications of computer vision such as feature tracking[13], image classfication[11], object detection[4], 2D and 3D registration[10,17]. A large number of applications promote various kinds of feature matching algorithms. At the same time, the continuous development of the applications put forward some new and higher requirements such as precision, speed and robust ability. In order to meet the needs of all these practical applications, much attention has been paid to improve the matching performance. A widely used method is computing variety of feature descriptors and select a threshold carefully to filter out large outliers. Therefore, variety of feature descriptors have been proposed, such as SIFT[8], SURF[9], BRISK[15], ORB[14] and LDB[19]. Further more, some people turned to combine more flexible geometric features and spatial characteristics. For instance, Chui et al.[5] introduced a feature based method named TPS-PRM (thin-plate spline-robust point matching) for non-rigid registration. C.Schmid[16] and Y.Zheng[20] use the thought of proximity. They assumed that two adjacent points in the original image should be matched to the couples which are also neighbours in the target image. X.Xu[18] use RANSAC and strong space constraints to obtain relatively stable feature point set first and then use a selection model[10] to decide which transformations are the most appropriate one. Finally, it constructs a global geometric

H. Zha et al. (Eds.): CCCV 2015, Part II, CCIS 547, pp. 190–200, 2015.
DOI: 10.1007/978-3-662-48570-5_19

transformation model as the matching constraint. O.Duchenne[6] accommodate both (mostly local) geometric invariants and image descriptors and search for correspondences by casting it as a hyper graph matching problem using higher order constraints.

Although many existing algorithms are general and could cover both rigid and non-rigid matching problems according to the problem definition, most of the them are either too computationally expensive to achieve real-time performance, or not sufficiently distinctive to identify correct matches from a large database with various transformations.

In this paper, we propose a tree structured hybrid feature matching algorithm, called DoG-based Random Grow (DoG-RG). In order to solve the problems mentioned above, we summarize our contributions as follows:

1. Flexible tree structure can effectively improve the patch-based discrimination of feature matching by combine feature lines with spatial distributions.
2. In order to increase the distinguish, we build a iterator method on the feature lines correspondences based on down sampling DoG pyramid.
3. Cell-space partitioning algorithm is used to reduce the selection number of candidate points in a limited area and which drastically speed up the matching process.

The remainder of the paper is organized as follows: Section 2 presents details of the proposed algorithm. In section 3, we compare performance of DoG-RG with some existing outstanding algorithms on public benchmarks. Section 4 gives the concluding remarks.

2 Algorithm Details

Suppose we have extracted two sets of feature points P^S and P^T from source image I^S and target image I^T. The overview of our proposed framework is shown in Figure 1. Firstly,we present the feature line extraction method; Secondly, we show the tree structure's start points(we call it anchors); Thirdly, we show the details of the exploring random tree(DoG-RG).

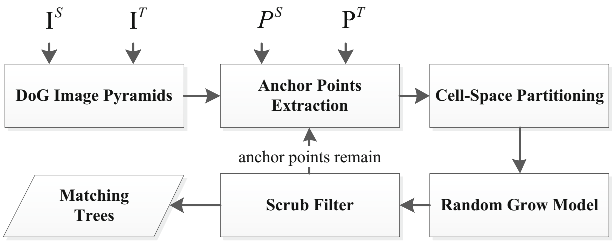

Fig. 1. The framework of the algorithm

2.1 Image Pyramids and Feature Lines

Considering the computational complexity, antinoise ability and characteristics of resolution, we use the DoG pyramid as the reference matching substrate. The lines between feature points are casted on the surface of the pyramid called feature lines.

Image Pyramids. We define L(x,y,δ) as a level in the multi-scale images and it is formed by a gaussian function G(x,y,δ) and an image I(x,y) convolution[12],described as follows:

$$L(x, y, \delta) = G(x, y, \delta) \otimes\ I(x, y) \tag{1}$$

We set $\otimes$ as the operator of convolution and G(x,y,δ) is:

$$G(x, y, \delta) = \frac{1}{2\pi\delta^2}\ e^{\frac{-(x^2+y^2)}{2\delta^2}} \tag{2}$$

We set σ as 1.5 and make a substraction between two adjacent scale-space images. The difference of gaussian image is denoted as D(x,y,δ). The function is given as follows:

$$D(x, y, \alpha) = S(k) * ((G(x, y, \sigma\delta) - G(x, y, \delta)) \otimes I(x, y) \tag{3}$$

where k is the down sampling factor, S is the down sampling function which is introduced to reduce the computational burden of feature lines's extraction and increase the robustness of feature lines.

Feature Lines. We cast the feature points onto the reference matching substrate of DoG pyramid through coordinate conversion. As is shown in Figure 2, the feature lines projected on the surface of substrate present different fluctuations. The higher the level of the substrate is, the more stable of its fluctuations will become. On the contrary, The lower level substrate is, the stronger the resolution of the feature line will be.

In order to construct feature lines, we sample discrete pixels normally along the corresponding spaced feature points from DoG images. The similarity evaluation can be expressed as the follow mathematical formula:

Let FL^S and FL^T be the sequence of points extract from different levels of DoG images and for any $P_i^S(x, y, z) \in FL^S$,$P_i^T(x, y, z) \in FL^T$. Then normalize the length of FL^S and FL^T as $N = max(length(FL^S),length(FL^T))$. Here similarity η is defined as

$$\eta = \frac{\sum\limits_{i=1}^{N-1} sign(P_i^S) \bigodot sign(P_i^T)}{N-1} \tag{4}$$

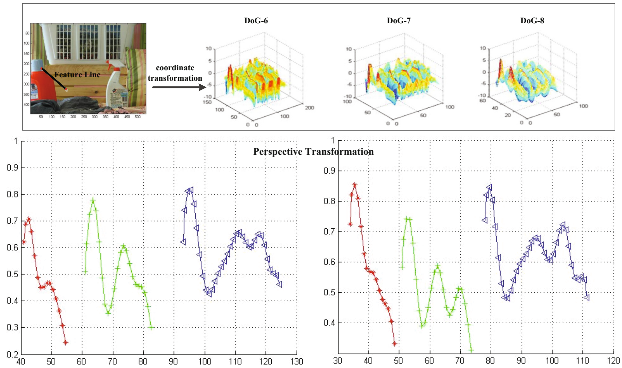

Fig. 2. An illustration of feature line extract from source and target image DoG-6:blue;green:DoG-7;red:DoG-8

where, $P_i.z$ means the gray value of DoG image and $sign(P_i)$ is defined as

$$sign(P_i) = \begin{cases} 1 \;, P_i.z - P_{i-1}.z > \varepsilon \\ -1 \;, P_i.z - P_{i-1}.z < -\varepsilon \\ 0 \;, abs(P_i.z - P_{i-1}.z) \leq \varepsilon \end{cases} \tag{5}$$

Where, the ε is a precision control factor given as follows, M is a feature line's resolution parameter.

$$\varepsilon = \frac{max(P_{set}.z - min(P_{set}.z))}{4M} \tag{6}$$

Generally, we set $N \geq 20$,otherwise, this feature line will be deemed invalid. More over, we set $\varepsilon = 0.7$, M = 13 and these settings are used in the subsequent experiments in this paper.

Anchor Points Extraction. In order to find more stable feature points (Anchor points), we establish a triangular structure, the start position of matching trees, which is constructed by three feature points and three feature lines. Based on the patch-based descriptors and more restricted similarities, we get small pieces of matched feature points. After that, several three-point combinations are randomly selected and checked by the spatial similarity and feature lines. If the triangular correspondence is wrong, the matching trees will always become low and will be filtered in the procedure Scrub Filter latter.

2.2 Tree Structured Random Grow Method

Space Partitioning. In order to speed up the matching process, we divide the feature points into grid cells according to space distribution. Assuming that feature points are under relatively uniform distributions, the division can contribute to avoiding a large number of outliers' operations.

For the subdivision, two basic principles are proposed as follows:

1. Minimize the number of points in each subdivision cell to improve the matching speed.
2. Retain enough cell size can increases the length of a feature line, strengthen the resolution and improve the matching accuracy.

As these two principles are contradictory, we propose an empirical formula. Suppose P^S's distribution area is $posW_S \times posH_S$, P^S size is $FeNum_S$,N is the minimum acceptable length of the feature lines. The division of cell number X_S can be calculated as:

$$X_S = min(\frac{\sqrt{posW_S * posH_S}}{2N}, \frac{\sqrt{FeNum_S}}{2}) \tag{7}$$

If the scale transformation happened, denoted by s, we can get the target point set P^T's division cell number X_T as:

$$X_T = min(\frac{\sqrt{posW_T * posH_T}}{2N}, \frac{\sqrt{FeNum_T}}{2}, \frac{X_S}{s}) \tag{8}$$

In the division, we put the feature points to the split cell grids. This strategy quite good to the quick index of feature points and exclude the outliers. Thus, It drastically increase the matching speed and improves the precision.

Random Grow Method. Suppose A^S is anchor point and let it as the root of a whole matching tree. Matching process starts from A^S and then search new points in the range of r^{branch} around. Assuming point D^S is belong to the range of r^{branch}, we use feature line $\overrightarrow{A^S D^S}$ and position relations to check the corresponding point D^T in the target point set P^T.

If the feature point D matching success, we set D^S as a new growing point and shrink the searching area to eight-neighborhood region. As shown in the Figure 3. In order to maintain the distinction of feature lines, our eight-neighborhood area ignore the center cell which is marked blue in the Figure 3. With the reference of the position D^S, we find out the corresponding position E^{vir} in I^T. Then we draw a circle (E^{vir}, r^{leaf}) and extract the contact cells, obtain all the candidates in the cells such as E_1^T and E_2^T. Finally, we use the descriptors and feature lines to sift out the best match. Traditionally, patch-based descriptors may hard to distinguish local repeat mode. Here,with the back-trace strategy of tree structure, we can easily find out local repetitive patterns and determine which one is the best. For example, if the feature lines from D^T failed to distinguish E_1^T and E_2^T, we just backtrack to A^T and build new feature lines to avoid passing through duplicate regions.

In the entire search process, we continually use the matched points to deduce the next points nearby till the end of matching process.

We continue the performs of algorithm till it can not find new anchor points to generate more matching trees. Since we can not ensure all the anchor points are correct matches, we have to filter the scrub to ensure a better accuracy.

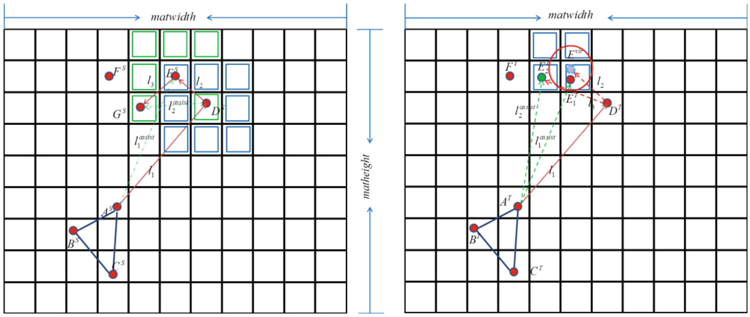

Fig. 3. An illustration of the matching process;The image at left reveals origin strategy and the one on the right shows the target

Due to the wrong matching trees don't match with the growing image region which always grow shorter, we can easily purify the matching trees by remove the scrubs.

3 Experimental Results

In the experiments presented here, we dived them into two parts: first we try to assess our method's actual capability on several image transform conditions(illumination, blur and compression). In order to guarantee the justice of experiment, we use OpenCV 2.4.6 to extract different size of feature points. To rule out the influence of patch-based descriptors, we repeat the experiments with the frequently-used descriptors: SIFT, SURF, ORB and BRISK.

Second, we give a set of images including local area transformation, irregular deformation and high repetitive pattern to show the good robustness of the algorithm.

3.1 Experiment Based on Different Local Feature Descriptors

Using the image groups of Tree, UBC and Leuven from data set [3], we exam the performance of blur, compress and light respectively. For each image group, the task is to match the first image to the remaining five, yielding five image pairs per sequence which are denoted as pair 1/2 to pair 1/6. Under the reference of the descriptors performance research [1], we carefully selected SIFT, SURF, ORB and BRISK as feature descriptors for the contrast experiments. To be fair, we compare DoG-RG with several the most frequently used feature descriptor combination algorithms integrated in OpenCV2.4.6, they are RADIUS, NNDR and BRUTE-FORCE. In order to verify the validity of corresponding

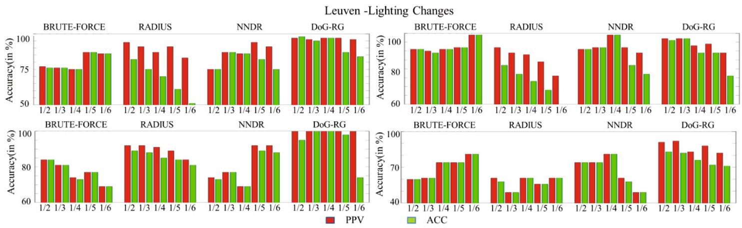

Fig. 4. PPV and ACC obtained by SIFT(Top left),SURF(Top right), ORB(Left bottom), BRISK(Right bottom) for the six image sequences of leuven

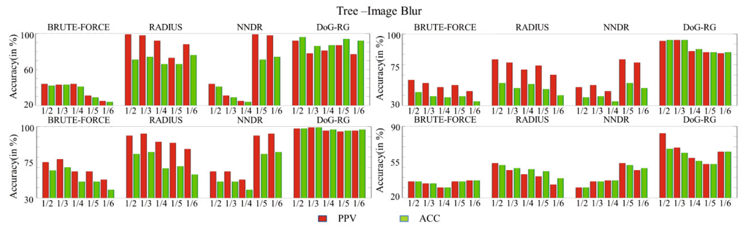

Fig. 5. PPV and ACC obtained by SIFT(Top left),SURF(Top right), ORB(Left bottom), BRISK(Right bottom) for the six image sequences of Trees

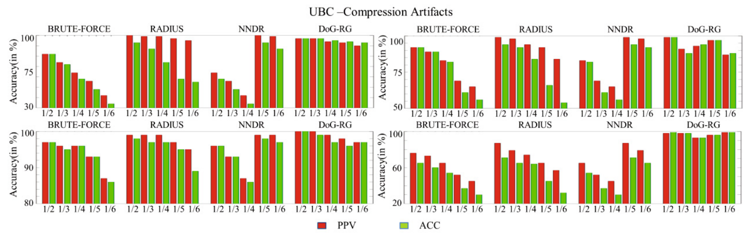

Fig. 6. PPV and ACC obtained by SIFT(Top left),SURF(Top right), ORB(Left bottom), BRISK(Right bottom) for the six image sequences of UBC

points, we use the combination of the above four algorithms to extract corresponding feature points. Correct matching enforces a one-to-one constraint so that a match is correct if two points are geometrically closet with sufficient overlap, and closest in feature space measure.

Two measures are introduced to evaluate the performances of all these methods according to the evaluation index ACC and PPV [7] and the calculating Formula 9 are listed below:

Table 1. Number of Feature Points Extracted

Sequence	Descriptor	1	2	3	4	5	6
Tree	SIFT	2613	2667	2940	3524	2163	2409
	SURF	1905	1872	1833	1637	1495	1424
	ORB	500	500	500	500	500	500
	BRISK	984	996	1033	995	818	573
Leuven	SIFT	861	682	574	521	433	349
	SURF	1313	1143	1036	937	802	650
	ORB	500	489	489	476	464	489
	BRISK	268	214	175	160	117	104
UBC	SIFT	1371	1348	1360	1418	1595	1597
	SURF	1602	1575	1620	1561	1582	1315
	ORB	500	500	500	500	500	500
	BRISK	546	558	507	503	571	774

Table 2. Threashold for matching.

Descriptor	RADIUS	NNDR	BruteForce	DoG-RG
SIFT	thr=0.24	ratio=1.0/1.2	thr=0.34	fl=0.6,DoG-level=8
SURF	thr=0.25	ratio=1.0/1.2	thr=0.35	fl=0.6,DoG-level=8
ORB	thr=65.0	ratio=1.0/1.1	thr=75	fl=0.6,DoG-level=8
BRISK	thr=145	ratio=1.0/1.1	thr=200	fl=0.6,DoG-level=8

$$\begin{aligned} Accuracy(ACC) &= \frac{TP+TN}{TP+TN+FN+FP} \\ Precision(PPV) &= \frac{TP}{TP+FP} \end{aligned} \tag{9}$$

Different feature points are extracted from sequence of test images, the points number are listed as Table 3.1.

We carefully select the thresholds for each patch-based method as shown in Table 2 by comprehensive considering of the overall performance. These settings are also used in DoG-RG's patch-based parts. Among them, fl is the feature line threshold, thr is the threshold of descriptors and $ratio$ is used for NNDR.

Through the experiment, we find that tree structured method obtains a quite higher performance than other algorithms. The explaining is that patch-based descriptors and feature lines are local, but the tree structured feature lines are more "global", this flexible structure enables us to overcome patch myopia. This strategy is attractive because of its simplicity and flexibility. The combination of feature descriptors and tree structured feature lines can effectively suppress the unstability of accuracy in different conditions and obtain more excellent results in general.

3.2 Matching in Irregular Deformation and Repeating Pattern

In order to further the ability test on irregular transformations, we select several common image transformations in real life .

1. The ability to match the partial translations and rotations in one scene.
2. Test matching capabilities under partial irregular deformation.
3. Test matching capabilities in high repeat patterns.

In the following cases, we combine the descriptor SIFT with dynamic feature lines to complete the matching process. Matching results are shown in the mosaic: the uppers are the original images, the middles are the feature matching trees and the lowers are the connections of the corresponding points.

Fig. 7. (UP)Local mobile origin images(Middle)Matching trees (Bottom)Matching figure

Our algorithm can easily handle the partial inconsistent deformations are seen in the Figure 7. Explanation is as follows: by using the tree-structured searching strategy, local gentle irregular deformation can be easily cope with local tree nodes searching strategy, regional steep deformations in different transformations can be easily solved by bring more different matching trees in.

These local irregular deformations of fisheye images are come from [2]. It can be seen from the Figure 8 that feature lines from high level Difference-of-Gaussians pyramid can effectively adapt to the local irregular deformations. Generally, this combination of feature descriptors and tree structured feature lines have a quite good robustness in irregular deformations.

Seen from Figure 9, although we do not use the consistent algorithm to purify the result, this proposed method can effectively distinguish the repeat patterns effectively. Even in dense points distribution area, dynamic feature lines strategy can still automatically select appropriate connections to achieve a good matching result. At the same time, feature lines can also prevent the spread of error matches. Just as the description in the figure, very few mismatched feature

Fig. 8. (UP)Fisheye origin images(Middle)Matching trees (Bottom)Matching figure

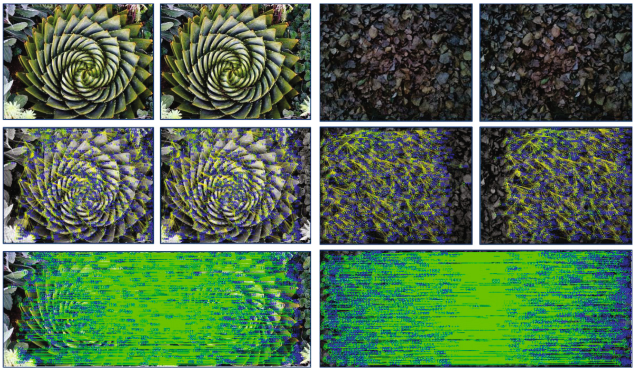

Fig. 9. (UP)Repetitive patterns (Middle)Matching trees (Bottom)Matching figure

points distributed in the border area are all isolated and these short trees will be removed in the scrub filter process.

4 Conclusion

In this paper, we propose a new image feature matching algorithm DoG-RG. By combining the feature lines with dynamic strategy and the patch-based feature descriptors, it constructs a incremental tree structured matching algorithm. The substantial benefits of this work is the good matching performance in simple calculation method and high robust ability. Experiment results show its better performance in common transformations and high local repetitive patterns. In addition, proposed methods can easily combine with various of patch-based descriptors to satisfy the needs of different matching conditions.

References

1. Comparison of feature descriptors. http://computer-vision-talks.com/2011/08/feature-descriptor-comparison-report/
2. Fisheye-hemi plug-in samples. http://imagetrendsinc.com/gallery/galleryhemi.asp
3. Visual geometry group-the oxford affine convariant regions datasets. http://www.robots.ox.ac.uk/vgg/research/affine/
4. Berg, A., Berg, T., Malik, J.: Shape matching and object recognition using low distortion correspondences. In: CVPR, vol. 1, pp. 26–33 (2005)
5. Chui, H., Rangarajan, A.: A new point matching algorithm for non-rigid registration. Computer Vision and Image Understanding **89**(2c3), 114–141 (2003). nonrigid Image Registration
6. Duchenne, O., Bach, F., Kweon, I.S., Ponce, J.: A tensor-based algorithm for high-order graph matching. PAMI **33**(12), 2383–2395 (2011)
7. Fawcett, T.: An introduction to roc analysis. Pattern Recognition Letters **27**(8), 861–874 (2006)
8. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision **60**(2), 91–110 (2004)
9. Bay, H., Tuytelaars, T., Van Gool, L.: Speeded-up robust features(surf). CVIU **110**(3), 346–359 (2008)
10. Izadi, M., Saeedi, P.: Robust weighted graph transformation matching for rigid and nonrigid image registration. IEEE Transactions on Image Processing **21**(10), 4369–4382 (2012)
11. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: spatial pyramid matching for recognizing natural scene categories. In: CVPR, vol. 2, pp. 2169–2178 (2006)
12. Lindeberg, T.: Feature detection with automatic scale selection. International Journal of Computer Vision **30**(2), 79–116 (1998)
13. Ong, E.J., Bowden, R.: Robust facial feature tracking using shape-constrained multiresolution-selected linear predictors. IEEE Transactions on Pattern Analysis and Machine Intelligence **33**(9), 1844–1859 (2011)
14. Rublee, E., Rabaud, V., Konolige, K.: Orb: an efficient alternative to sift or surf. In: ICCV, pp. 2564–2571 (2011)
15. Leutenegger, S., Chli, M., Siegwart, R.Y : Brisk: binary robust invariant scalable keypoints. In: ICCV, pp. 2548–2555 (2011)
16. Schmid, C., Mohr, R.: Local grayvalue invariants for image retrieval. PAMI **19**(5), 530–535 (1997)
17. Wu, Z., Goshtasby, A.: Adaptive image registration via hierarchical voronoi subdivision. IEEE Transactions on Image Processing **21**(5), 2464–2473 (2012)
18. Xu, X., Yu, C., Zhou, J.: Robust feature point matching based on geometric consistency and affine invariant spatial constraint. In: ICIP, pp. 2077–2081 (2013)
19. Yang, X., Cheng, K.T.: Local difference binary for ultrafast and distinctive feature description. IEEE Transactions on Pattern Analysis and Machine Intelligence **36**(1), 188–194 (2014)
20. Zheng, Y., Doermann, D.: Robust point matching for nonrigid shapes by preserving local neighborhood structures. PAMI **28**(4), 643–649 (2006)