Pacific Graphics (2018)
H. Fu, A. Ghosh, and J. Kopf (Editors)

Extreme Feature Regions for Image Matching

Baijiang Fan[1], Yunbo Rao[1], Jiansu Pu[1] and Jianhua Deng[2]

[1] School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu, China;
[2] School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, China;

Abstract

Extreme feature regions are increasingly critical for many image matching applications on affine image-pairs. In this paper, we focus on the time-consumption and accuracy of using extreme feature regions to do the affine-invariant image matching. Specifically, we proposed novel image matching algorithm using three types of critical points in Morse theory to calculate precise extreme feature regions. Furthermore, Random Sample Consensus (RANSAC) method is used to eliminate the features of complex background, and improve the accuracy of the extreme feature regions. Moreover, the saddle regions is used to calculate the covariance matrix for image matching. Extensive experiments on several benchmark image matching databases validate the superiority of the proposed approaches over many recently proposed affine-invariant SIFT algorithms.

CCS Concepts

•***Computing methodologies*** → *Image processing; image-matching; random sample consensus; affine invariant;*

1. Introduction

Local invariant feature detector can occupy very important position in many research areas, such as multiple images point cloud reconstruction, face tracking and object recognition [SYL*17]. Traditional feature detector searches interest points with high curvature, which includes corner or edge [WS16]. SIFT is a state-of-art feature matching method, which can match features in different scale spaces. However, the accuracy of SIFT can seriously decline when handling with images under different view perspectives [Slu15]. Rosten E et.al [ERT09] proposed to use machine learning on corner feature detecting and achieved a high processing time of 5% of per frame. This method needs a long time to build the descriptor. Also, this method gets unsatisfied results when matching corners on affine image pairs, because the descriptor of corner changes under different view perspectives.

Yu et.al [YM09] proposed a fully affine-invariant SIFT (ASIFT) method. It can retrieve the object under extreme change of angle if the object has rather flat surface. ASIFT builds a hemisphere space and uses two parameters: longitude and latitude to find the position of camera, then imitates the affine transformation. However, ASIFT is highly complex. The time complexity of ASIFT is nearly 6 times of SIFT in extreme cases [BJU10]. Another fully affine-invariant feature detecting method is the Maximally Stable Extreme Region (MSER) [XGN16]. This method works by creating max-tree and min-tree called MSER+ and MSER- of the gray-scale images. And uses a stable extract function to find node regions in the image and finally get root regions along the successive nodes area. MSER can simply expressed as :

$$S = (E_{\Delta-} - E_{\Delta+})/E \tag{1}$$

Where S is the MSER result regions, $E_{\Delta-}$ and $E_{\Delta+}$ are the max-tree and min-tree. The complexity of MSER is $o(nlog(log_n))$, which is nearly achieve linear [JOMT04]. MSER also has high accuracy and achieved state-of-art method. Recently, some researchers proposed to combine the advantages of MSER and SIFT. Hu et.al [HZG17] proposed to find MSER regions on image-pairs with different view angles and use these regions to do SIFT matching. However, the number of regions tracked by normal MSER is small, and MSER regions always contain many useless backgrounds, which decreases the accuracy and increases time consuming. Torr et.al [TZ00] tries to normalize the affine image-pairs by MSER. This method is based on the theory that affine transform matrix can be derived by three groups of matched points. However, this method cannot find enough results in many situations.

In our work, we propose a precise extreme feature region (PEFR) method for image matching. The main contributions including: (1) we improve the tree structure of MSER method to increase the number of precise extreme regions, and eliminate the feature of complex background regions according to the new characters of PEFR regions. (2) we calculate the affine matrix by the saddle points and use SIFT detector on each normalized region to get matching result.

2. The Proposed Method

In this section, we describe the proposed method details. The flowchart of our fusion method is shown in Fig.1. It consists of the following four steps: Morse-and-MSER-based PEFR region, background elimination, affine normalization, and SIFT matching.

DOI: 10.2312/pg.20181286

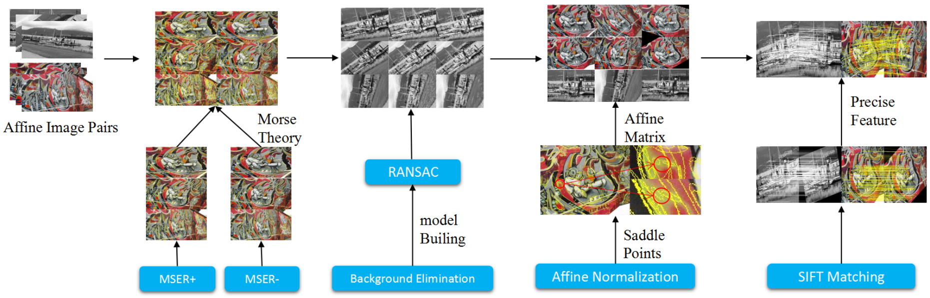

Figure 1: *the flowchart of proposed method.*

2.1. Morse-and-MSER-based PEFR region

Morse theory aimed at describe the topological changes of the color level in image in terms of the critical points. Morse theory works on smooth real-valued manifold to find critical points. Critical point include minimum points, maximum points and saddle points. Critical points give clues to research the topological relationship of the level in a local space, however this method can only work on non-degenerate areas and can not track flat area. Meanwhile, it is not enough for tracking feature regions from image.

In our work, we use MSER tree structure to simulate the non-degenerate situation of Morse theory and find the tree structure. For minimum and maximum points, MSER tree-structure can get results as Fig.2.

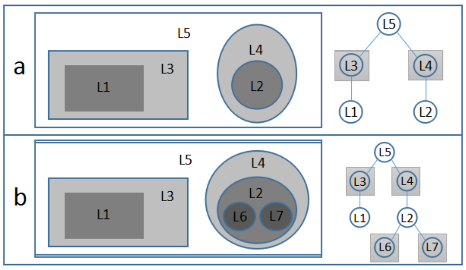

Figure 2: *tree-structure of feature region.*

In Fig.2(a). $L1 \in L3, L2 \in L4, L3\&L4 \in L5$. MSER method only keeps L5 in regions list. Here we define the minimum region and maximum region as extreme region:

$$\exists R\&R' \supset R, \forall r \subset R', r \cup R = \emptyset \rightarrow R = ExtremeR \qquad (2)$$

Here we keep extreme regions L3 and L4 in our region list. L1 and L2 are more precise regions than L3 and L4, which has the same topological means as L3 and L4. However, it is just the extension of the father nodes, not the leaves. In order to avoid losing features, we choose the extreme regions L3 and L4 to keep more space. On the other hand, L3 and L4 are the separate of a node, which means different areas in topological conception.

The number of feature will depend on the layer of the tree. As Fig.2(b), $L6\&L7 \in L2, L3\&L4 \in L5$, so extreme regions L6, L7 and L3, L4 will be kept in the PEFR region list. It will bring an exponential growth on the number of results, which makes the high time consuming when we can get many layers in a MSER tree. In our work, a max layer value 4 is used.

Saddle points finding by topological tree is critical problem. In our work, a contour-based method is used to find the saddle regions. A saddle area can exist at the contact position of several PEFR regions, as Fig.3 (b) shows. Our method finds the area where several PEFR regions contact and keeps them as saddle regions.

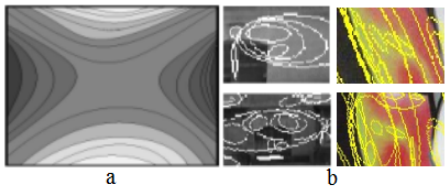

Figure 3: *contour map of gray-scale. (a) is a contour map of gray-scale, four images in (b) is the contour map by PEFR.*

PEFR method can also finds flat areas. Morse theory cannot find the flat areas due to flat areas always keep the same gray-scale, MSER tree-structure can tract flat area easily. However, in MSER, flat areas may be covered by higher gray-scale. In our method, we keep the regions which not grow when the threshold increases. These areas are the flat regions. Flat areas will be calculated at last, in case of the influence to the tract of saddle regions.

2.2. Background elimination

The proposed PEFR method brings regions new characters to separate regions in background from the foreground. The background elimination can decrease the time consumption in the following steps. RANSAC method is used to eliminate the regions in background as algorithm.1 shows. RANSAC works as a filter by iterative calculation. We build a region detect model to separate background from foreground.

Fig.4 (a) and Fig.4 (b) are the bright regions and dark regions of PEFR method. We find that after precise curving, regions in foreground become more focus and have more layers, but the background are more disperse and have lower layers or even have only one layer. According to this characteristic, we use three thresholds

Algorithm 1 Background elimination RANSAC

Input: The set of PEFR regions, R; The threshold, X, Y, M;
Output: Boolean result, *result*;
$r = R.radius; R = R.around\_regions; r1 = R.radius$;
if $r > X$ **then** result = 1;
else if r1 <= X and r1 >= Y **then** result = 0;
else if r1 >= X/3 and R.amount <= M **then** result = 0;
end if

X, Y and M to eliminate the background. X is the first radius threshold to compare with the radius of each region. Y is the second radius threshold to compare with the around regions of some regions. M is the third threshold to compare with the amount of the around regions. We manually choose some foreground regions and some background regions to calculate the parameters X, Y and M.

Figure 4: *background of PEFR, (a) is the bright_PEFR, (b) is the dark_PEFR. Regions in the rectangle are background regions. Comparing with regions in the house, they are disperse and small.*

2.3. Affine Normalization

Affine transformation is the combination of linear transformation (enlargement, reduction and rotation) and translation. In order to build the descriptor of the saddle points, we propose method fusing the centroid of a MSER region and the relative gradient of the saddle point. We choose soble detector to calculate the gradient. As Fig.5 shows, we use the location and rotation of the saddle point as the descriptor. Intersection angle α of L1 and L2 is:

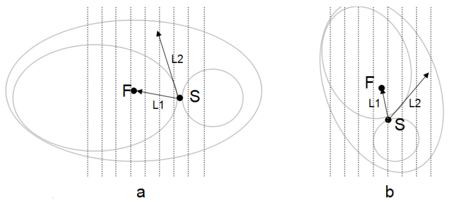

Figure 5: *relationship between saddle and centroid. Regions in (b) is the affine situation of regions in (a).*

$$\alpha_n = arctan\frac{|k_{n1} - k_{n2}|}{|1 + k_{n1} * k_{n2}|}(k_{n1} * k_{n2} \neq 0) \tag{3}$$

where $k_{n1} = \frac{d_x}{d_y}, k_{n2} = \frac{|S_{nx} - F_{nx}|}{|S_{ny} - F_{ny}|}$, S and F are the saddle and the centroid. If $k_{n1} * k_{n2} = -1, \alpha = 90^{\circ}$. We build the relative gradient as (5),

$$T_n = \sqrt{T_{nx}^2 + T_{ny}^2} \tag{4}$$

where $T_{nx} = \frac{f(x,y+Rcos\beta) - f(x,y-Rcos\beta)}{R}$, T_{ny} is as the same formula, and $\beta = arctank_{n1}$.The saddle will be described as a set of intersection angle and relative gradient Tn. We store saddle descriptors of each image in a list, and then match the saddle by the descriptor. We choose multi-groups of saddles to calculate the affine matrix, which can eliminate the match error.

2.4. Image matching stage

SIFT will match the feature in the normalized PEFR, to achieve the affine invariant SIFT method. Original SIFT detector is complex. However, our proposed PEFR only have simple logical cutting and normalization, and has the same time complexity as MSER of $o(n)$. Our method does not increase the efficiency of SIFT, besides, the background elimination even decrease the running time.The proposed PEFR method can be conclude as algorithm.2:

Algorithm 2 Progress of PEFR Algorithm

Input: The input image, I; The set of extremal regions, R; The threshold, T; The set of saddle points, S;
Output: The list of PEFR regions, P;
$T = 0$, $R =$ extremal region (I, T);
while (T != 255) **do**
    R = get region(I,T++);
    R = get region(reverse(I),T++);
end while
if $(R.i + R.j) > (|R.i| + |R.j|)$ **then**
    P.add(extreme region($R.i$,$R.j$));
end if
Eliminate(Ransac($P.i$))
S.add(Contour($P.i$, $P.j$))
Affine-Matrix = match(S);
return SIFT($P*$ Affine-Matrix);

3. Experimental Results

We display a quantity of results of important steps of our method in this section. The proposed algorithm will be implemented on

Figure 6: *matching result by our method. (a) is the result on normalized image, (b) is the result on original image.*

OpenMVG framework and use Oxford-VGG dataset. First we use the affine matrix to repair the affine image and uses SIFT to match the feature points. Then we use the matrix to get the original position of the matched features and get another result. Fig.6 is the matching result of our method. Fig.6(a) is the result with normalized image, Fig.6(b) is the result with original image.

We noticed that our method can get precise matching features on affine images, and most of the features are focus on the foreground part. In Fig.7 we compare our method with SIFT and Hu's [HZG17] method which combines normal MSER with SIFT.

Figure 7: *comparison of SIFT and PEFR. (a) is result of SIFT, (b) is result of Hu's method, (c) is result of our method.*

Figure 8: *comparison with [ERT09]. (a): our result on low affine image-pair and high image pair. (b),(c): FAST-9 and FAST-ER on low affine and high affine.*

Table.1 compared the accuracy, feature number and time costing of SIFT, ASIFT [BJU10], Hu's method [HZG17] and our method on different inputs. In Fig.8 we also compare our method with Edward's methods [ERT09] which using machine learning to detect corner features.

Table.2 is the comparison of result on low affine and high affine image-pairs. From table.1 and table.2, we find our result better than SIFT in feature number and time costing and better than Hu's [HZG17] method in accuracy and feature number. A-SIFT can get more matches, but it is costs much more time than our method. Edward's methods can only get few match points, especially when handle high affine image-pairs. Our method costs a little more time because it need more time to test the accuracy of features.

Table 1: *Comparison with other methods*

		SIFT	ASIFT	Hu's method	Our method
GRAF	Accurate(%)	92.3	91.1	85.7	95.1
	feature number	45	2765	110	405
	running times(s)	0.9885	209	0.7574	0.872
SHIP	Accurate(%)	96.7	96.1	91.4	96.5
	feature number	86	1757	221	609
	running times(s)	1.1994	284	0.782	0.9681

Table 2: *Comparison with machine learning*

		FAST-9	FAST-ER	Our method
Low Affine	feature number	2601	2765	405
	running times(s)	206	209	0.872
High Affine	feature number	86	1757	609
	running times(s)	1.1994	284	0.9681

4. Acknowledge

This work was supported by the Fundamental Research Funds for the Central Universities of China (No.A03013023001050, No.ZYGX2016J095). Natural Science Foundation of Sichuan (No.2017JY0229). CERNET Innovation Project(NGII20170805).

References

[BJU10] Bresе, Christopher Le and Ju, Jia Zou and Uy, Brian: An improved ASIFT algorithm for matching repeated patterns. *IEEE International Conference on Image Processing*, (2010), 2949–2952. 1, 4

[ERT09] Edward Rosten and Reid Porter and Tom Drummond: Faster and better: a machine learning approach to corner detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32 (2009), 105–119. 1, 4

[HZG17] Hu, Wenchao and Zhou, Wei and Guan, Jian: A modified M-SIFT algorithm for matching images with different viewing angle. *IEEE International Conference on Signal and Image Processing*, (2017), 557–560. 1, 4

[JOMT04] J Matas and O Chum and M Urban and T Pajdla: Robust wide-baseline stereo from maximally stable extremal regions. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 22 (2004), 761–767. 1

[Slu15] Sluzek, Andrzej: Contextual descriptors improving credibility of keypoint matching: Harris-Affine, Hessian-Affme and SIFT feasibility study. *International Conference on Control Automation Robotics & Vision*, (2015), 117–122. 1

[SYL*17] Shih, Ya Fang and Yeh, Yang Ming and Lin, Yen Yu and Weng, Ming Fang: Deep co-occurrence feature learning for visual object recognition. *IEEE Conference on Computer Vision and Pattern Recognition*, (2017), 7302–7311. 1

[TZ00] Torr, P. H. S and Zisserman, A: MLESAC: a new robust estimator with application to estimating image geometry. *Computer Vision & Image Understanding*, 78 (2000), 138–156. 1

[WS16] Wang, Fu Ping and Shui, Peng Lang: Noise-robust color edge detector using gradient matrix and anisotropic Gaussian directional derivative matrix. *Pattern Recognition*, 52 (2016), 346–357. 1

[XGN16] Xu, Yongchao and Geraud, Thierry and Najman, Laurent: Connected filtering on tree-Based shape-Spaces. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (2016), 1126–1140. 1

[YM09] Yu, Guoshen and Morel, Jean Michel: A fully affine invariant image comparison method. *IEEE International Conference on Acoustics, Speech and Signal Processing*, (2009), 1597–1600. 1