Robust Non-Parametric Template Matching with Local Rigidity Constraints

Chao ZHANG\(^{(a)}\), Student Member, Haitian SUN\(^{(b)}\), Nonmember, and Takuya AKASHI\(^{(c)}\), Member

SUMMARY In this paper, we address the problem of non-parametric template matching which does not assume any specific deformation models. In real-world matching scenarios, deformation between a template and a matching result usually appears to be non-rigid and non-linear. We propose a novel approach called local rigidity constraints (LRC). LRC is built based on an assumption that the local rigidity, which is referred to as structural persistence between image patches, can help the algorithm to achieve better performance. A spatial relation test is proposed to weight the rigidity between two image patches. When estimating visual similarity under an unconstrained environment, high-level similarity (e.g. with complex geometry transformations) can then be estimated by investigating the number of LRC. In the searching step, exhaustive matching is possible because of the simplicity of the algorithm. Global maximum is given out as the final matching result. To evaluate our method, we carry out a comprehensive comparison on a publicly available benchmark and show that our method can outperform the state-of-the-art method.

key words: non-parametric template matching, local rigidity constraints, visual similarity estimation

1. Introduction

Template matching has been studied as a classical problem for a number of decades. Current techniques can match template with similar candidate windows in the target image while estimating translation, rotation, affine transformation and even some regular deformation. However, in real-world applications such as online tracking [1], [2], the foreground models in the templates usually deform complexly. Such deformation can hardly be modeled mathematically since the result in the target image is projected from the template after involving fusion of 3D transformations. Furthermore, external influences, such as occlusion, illumination change, and background clutter will increase the degree of non-linearity and the difficulty of template matching.

The drawbacks of existing parametric template matching frameworks can be mainly summarized as: 1) Dense matching at pixel-level is usually necessary in order to estimate more accurate parameters, which requires a lot of computing costs; 2) In the case of heavy occlusion, the occluded part may drastically affect the whole similarity score and lead to mismatch; 3) A large number of parameters may need to be estimated when complex transformations occur, which is very difficult due to high-dimensionality and strong non-convexity. These drawbacks limit the scope of applications and increase the dependence on environment.

As a common solution, histogram matching (HM) plays an important role in non-parametric template matching. HM can deal with the deformable matching problem by disregarding geometric relationship between pixels. However, we argue that disregarding geometry completely may increase the number of local optimums and thus increase the level of non-convexity. When partial occlusion occurs to the target object, the template is also easy to be mismatched to a local optimum [3].

In this paper, we address the importance of local rigidity constraints. As a circle can be approximated by many rigid straight lines, most of the objects can be approximated by rigid patches if the size of each rigid patch is small enough. Figure 1 shows an example of matching result with an athlete. We specify a region of interest (ROI) as a template in the reference image which is taken at a running race. As the race progresses, the appearance of the athlete continuously changes. By decomposing the template and the result into 3\(\times\)3 patches, we can find that although the global appearance has been changed (e.g. hands, head, legs and background), many patches still have corresponding relationships between the template and the result. Two patches and their corresponding relationship

Fig. 1 Matching example and its local rigidity constraints (LRC) which are generated by our program. Template area is predetermined within a reference image. Result area is represented by red rectangle, ground truth is presented by green rectangle in both target image and likelihood image. Dotted lines represent the corresponding LRC with 3\(\times\)3 pixel rigid patches.
As the most important component in template matching, cross-correlation (NCC), and zero-means normalized cross-correlation (ZNCC) have been widely applied. Due to dense matching, each cube of result in the target image changes its appearance compared to the template. Hence, 3D rigidity is more difficult to be preserved in 2D images.

2.1 Visual Similarity Estimation

In this paper, we mainly focus on the problem with single template and single result. Cross-domain matching is not considered in this paper (e.g. template is a sketch image while result is an RGB image). We survey previous works and contributions in this section. In Sect. 2.1, we conclude the basic distance measurement methods of similarity. These methods are widely applied in the applications of template matching, which will be summarized in Sects. 2.2 and 2.3 respectively in terms of using geometry model or not using.

2.2 Parametric Template Matching

Lucas et al. [6] proposed parametric optical flow to estimate inliers between a template and a target. Further developed by feature-based methods, Lucas and Kanade’s framework has become an essential approach in many matching problems. Combined with RANSAC [12], parameters of transformation can be estimated. However, a sufficient number of inliers (i.e., distinct features) are necessary in order to estimate the parameters. On the other hand, although affine or projective transformation can be calculated by solving a system of linear equations, parameters of non-rigid transformations are difficult to be calculated. C. Zhang and T. Akashi [7] proposed a stochastic method to search the 2D affine parameters efficiently with the fitness function of SAD. It is also difficult to be applied in real images since the real-world transformations are more complex. D.J. Tan et al. [13] modelled the 2D deformation with cubic B-Splines. Larger number of control points are required if more complex deformation want to be matched. Overall, to the best of our knowledge, most of the parametric methods can hardly be applied to “wild” images which may contain calculable and unpredictable deformation.

2.3 Non-Parametric Template Matching

It is more efficient and feasible to design deformation-invariant features instead of estimating the specific parameters when matching with the real-world deformation. Despite histogram matching [14]–[16], D.P. Huttenlocher et al. [17] designed a distance function which measures the level of mismatch between two point sets. The distance is calculated when point of set A is the farthest from any point of set B and vice versa. This idea is quite similar with [4]. The difference is, instead of calculating the farthest distance, [4] counts the number of matches which satisfy the condition: point of set A is the nearest from any point of set B and vice versa. Y. Rubner et al. [18] introduced a function named Earth Mover’s Distance (EMD) to measure the minimum cost that must be paid from one point set to another. EMD allows partial matches, which means it is robust with occlusion and clutter. D. Simakov et al. [19] proposed bidirectional similarity (BDS). BDS considers two point sets are similar if all patches of set A are contained in especially dealing with rotation, scaling, and simple transformation. However, feature-based methods are usually high-dimensional (e.g. SIFT is typically 128-dimensional), it is inefficient for dense matching and is usually employed with key points. It has also been discussed in [11] that the most damaging effect on the matching results of keypoint-based local features are the non-planarities or non-rigid deformation, which abound in our testing images. To cover the advantages of both direct method and feature-based method, Dekel et al. [4] found an intrinsic relationship between two groups of points, thus can reduce the outlier rate without increasing the number of feature dimension.
set B and vice versa.

It is worth pointing out that the term “local rigidity constraint” has also been used by [20] and related image registration papers. However, the definition is quite different with our method since in [20], local rigidity constraint is treated as a penalty term of the cost function, which is based on Jacobian matrix. In our paper, LRC is treated as a one-to-one map limitation (i.e. restriction) between a rigid patch of template and a rigid patch of candidate.

3. Methodology

Two color images are given as the input with each pixel and each channel normalized to [0, 1]. $I_T$ is defined as $T_w \times T_h$ pixel template image extracted from a reference image and $I_S$ is defined as a $S_w \times S_h$ pixel target image (i.e. source image). Each rigid patch is defined as a $s \times s$ pixel square patch. $T_w, T_h, S_w, S_h, s \in \mathbb{N}^+$. A candidate $I_C$ in target image $I_S$ is an ROI defined by a search window. Following the traditional sliding window search method, we have candidates arranged in order from top left to bottom right in the target image. To clarify the meaning of reference, target, and template image, we define them as following:

**Reference image**: Base image from which a template is cropped.

**Template image**: A region cropped from the reference image manually, and holds semantic meaning (usually an object) for matching with similar objects in the target image.

**Target image**: Also known as source image, is an image in which the object described by template exists, but may change in appearance due to internal and external influences.

**Problem** The problem of this paper can be defined as:

$$\arg \max_{I_C \in I_S} \text{LRCS}(I_T, I_C),$$

where LRCS($I_T, I_C$) is a function to estimate the LRC similarity between a template and a candidate, which will be introduced in the following section.

3.1 Feature of Local Rigid Patch

Feature vector of a rigid patch is denoted by:

$$f = p^{(L,C,G)}, \quad f \in \mathbb{R}^{6x^2+2},$$

where $L$, $C$, $G$ are feature spaces. $p^L \in \mathbb{R}^2$ denotes a patch’s center location in image $I$. Specifically, $p^L_x$ represents x-axis value and $p^L_y$ represents y-axis value. $p^C \in \mathbb{R}^{k \times 3}$ denotes a patch’s color feature (e.g. RGB). $p^G$ represents a patch’s spatial structure. The dimensionality of $p^G$ depends on the presentation of spatial structure. In this paper, $p^G \in \mathbb{R}^{k \times n \times n}$.

**Definition 1** The operator to investigate a neighbor pixel’s feature value is denoted by $[\cdot]_{x,y}$. $x, y$ is the relative coordinate to the location of corresponding pixel. Specifically, when $x = 1, y = 1$, and $p^C_i$ is the feature value obtained at $(2, 2)$ in the image coordinate, then $[p^C_{x,y}]_{x,y}$ equals to the feature value of the same dimension $i$ which is located at $(3, 3)$.

With Definition 1 in mind, we represent a patch’s spatial feature by investigating the relationship between each pixel in the patch and its neighbor pixels.

$$p^G_i := ||p^C_{i_{-1,0}} - p^C_{i}| - \omega|T_{x,s}, 0||/8 + ||p^C_{i_{-1,0}} - p^C_{i}| - \omega|T_{x,s}, 0||/8 + ||p^C_{i_{-1,0}} - p^C_{i}| - \omega|T_{x,s}, 0||/8,$$

where $\omega$ is a linear function to dynamically determine the neighborhood pixels to investigate according to the template’s size. Instead of fixing the position of neighborhood pixels to investigate (e.g. local binary pattern), we change the position dynamically based on a simple fact: the size of rigid parts depend on the size of object in the template.

The design of $p^G$ is important since we can confirm how the complex deformation affects the patches by checking the spatial structure feature in both template and candidate images. To gain more insight into the feature design, we refer to a simple case when $s = 3, C = RGB, \omega(\cdot) = 1$. Figure 3 illustrates this case.

**Definition 2** The operator to calculate the feature distance between two rigid patches is denoted by $||\cdot||_2$, which is defined as:

$$||f_1, f_2||_2 : \mathbb{R}^{6x^2+2} \times \mathbb{R}^{6x^2+2} \rightarrow \mathbb{R}^+: f_1, f_2 \mapsto w_L||p^L - q^L||_2^2 + w_C||p^C - q^C||_2^2 + w_G||p^G - q^G||_2^2,$$

where $f_1$ and $f_2$ are the feature vectors of two rigid patches. $w_L$, $w_C$, $w_G$ are the weights of each feature space. These weights balance the feature space to describe better appearance model of a template. In the experiment of parameter analysis, we will comprehensively study how the $w_L$, $w_C$, and $w_G$ affect the performance (each of them is varied from 0.5 to 3.0).
3.2 Local Rigidity Constraint Similarity

With the distance between two feature vectors defined, we can estimate the similarity between two rigid patches. Following traditional method such as SAD, we may estimate the similarity between template and candidate by using the sum of patches’ distance. However, it has been proved to be inefficient when deformation occurred in the target image and the corresponding relationships between pixel pairs no longer exist. Instead of using the feature distance to estimate the distance directly, we extend the method in [4].

Definition 3 The operator to judge whether constraint exists between a rigid patch of template image $I_T$ and a rigid patch of candidate image $I_C$ is denoted by $<\cdot, \cdot>_{I_T, I_C}$, which is defined as:

\[
<f_1, f_2>_{I_T, I_C} : \mathbb{R}^{6x^2+2} \times \mathbb{R}^{6x^2+2} \rightarrow \{0, 1\} : f_1, f_2 \mapsto
\begin{cases} 
1, & \text{NN}(f_1, I_C) = f_2 \land \text{NN}(f_2, I_T) = f_1 \\
0, & \text{otherwise},
\end{cases}
\tag{5}
\]

Where $\text{NN}(f_1, I_C) = \arg \min_{f \in I_C} \|f\|_2$, and $\text{NN}(f_2, I_T) = \arg \min_{f \in I_T} \|f\|_2$. Both $f_1$ and $f_2$ are feature vector of single patch which is extracted from $I_T$ and $I_C$ respectively. Similar operator is also defined in [4]. This operator is similar with binary quantization, which converts a pair of feature distance in real number into a countable number. This operator can also be seen as a compression procedure. By summing up 0/1 number, the degree of similarity can be compressed from high-dimensional feature space.

Definition 4 The LRC similarity between a template and a candidate can then be defined as:

\[
\text{LRCS}(I_T, I_C) : \mathbb{R}^{T_a \times T_b}, \mathbb{R}^{T_a \times T_b} \rightarrow \mathbb{R}^+: I_T, I_C \mapsto \frac{1}{\min(\|f\|_1, \|f\|_2)} \sum_{i,j} <f^T_i, f^C_j>_{I_T, I_C},
\tag{6}
\]

where $f^T_i$ is the feature vector of $i^{th}$ rigid patch extracted from $I_T$, $f^C_j$ is the feature vector of $j^{th}$ rigid patch extracted from $I_C$, $f^T = [f^T_1, f^T_2, \ldots]$, $f^C = [f^C_1, f^C_2, \ldots]$. Overall, Eqs. (5) and (6) specify the general expressions in Eqs. (1) and (2) of [4]. Our contribution is to add a spatial relation test defined in Eq. (3) to feature extraction, which helps to match rigid patches. The whole procedure of our algorithm is concluded in Algorithm 1.

3.3 Discussion

In mathematics and physics, the definition of “rigidity” can also be referred to as “stiffness”, which means the property of a solid body to resist deformation. In our paper, the “rigidity” has the similar meaning, which means the property of an image patch to resist geometry deformation. Furthermore, as each image patch is locally existed with respect to an image, we name it as “local rigidity”. Let us show a specific situation to visually illustrates the difference between BBS and LRC in Fig. 4. As we can see from Fig. 4, by involving such a certain pattern of spatial relation test, LRC tends to match patches that are structurally persistent. From this example, it is hard to judge directly whether the matching result of LRC is better than BBS. However, the result of LRC is more in line with our assumption: the matched image patches which are structurally persistent (rigid) play more important role on similarity estimation, and both the symmetric and real-data experiments show that this assumption can help improving the performance.

3.4 Analysis

In order to understand the efficiency, we first show a simple 2D case which is illustrated in Fig. 5. To increase the matching difficulty, two different background models (in red points) are generated, and each of them is mixed with the foreground model (in blue points). We match (a) and (b) and compare the results generated without LRC (c, d) and with LRC (e, f). By comparing with result (c) and (e), we can see that the number of matched foreground points is roughly doubled while the number of matched background points only increased by 12. The proportion of matched foreground points increases from 57% to 69%. By comparing result (d) and (f), we can also find that the number of matched foreground points is roughly doubled while the number of matched background points only increased by 7. The proportion of matched foreground points increases from 65% to 75%. This example shows that considering LRC can further improve the matching rate of foreground and separate the background well comparing with [4].

To prove LRC as a better method, we have to prove two assumptions: 1) The expectation of a pair of rigid patches to be matched is highest when two patches are from the same foreground (same distribution). Conversely, the expectation drops sharply when two foreground models leave each other. 2) If rigidity exist, considering the neighbor patches can enhance the phenomenon described in (1). We prove these two assumptions under one-dimensional case. First we generate a point set $P$ under normal distribution $N(0, 0.1)$, $\|P\| = 100$, and then extend $P$ to $\overline{P} = \{P, P - d, P + d\}$, $d \in \mathbb{R}, \|P\| = 300$. Similarly, we generate $\overline{Q}$ under $N(\mu, \sigma)$ and extend it to $\overline{Q} = \{Q, Q - d, Q + d\}, \|\overline{Q}\| = 300$. Note

\begin{algorithm}[h]
\caption{Template matching with LRC.}
\begin{algorithmic}[1]
\Require Template image extracted from reference image: $I_T$
\Require Target image: $I_C$
\Require Size of rigid patch: $s$
\For{$i$ from 1 to $T_a$}
\For{$j$ from 1 to $T_b$}
\If{$i - s/2 \geq 0 \land j - s/2 \geq 0 \land i + s/2 \leq T_a \land j + s/2 \leq T_b$}
\State Preprocessing with Gaussian smoothing
\State Calculate LRCS($I_T, I_C$), the center of $I_C$ locates at $(i, j)$
\EndIf
\EndFor
\EndFor
\State Return $\arg \max_{IC \in IS} \text{LRCS}(I_T, I_C)$
\end{algorithmic}
\end{algorithm}
that the points in \( \mathcal{P} \) and \( \mathcal{Q} \) no longer obey simple Gaussian distribution since \( P - d, P + d, Q - d, Q + d \) are involved. The expectation of two points \((p, q), p \in \mathcal{P}, q \in \mathcal{Q}\) to be matched can be defined as \( E_1 \). The case in which \( \mathcal{P} \) and \( \mathcal{Q} \) are simple Gaussian distributions has been proved in [4]:

\[
E_1 := \int_{-\infty}^{\infty} \left( f_{\mathcal{P}}(p)F_{\mathcal{Q}}(|x - p| \leq |p - q|)\right)^{-1} \times f_{\mathcal{Q}}(q)F_{\mathcal{P}}(|x - q| \leq |p - q|)\|\mathcal{P}\|^{-1} dp dq, \tag{7}
\]

where \( F_{\mathcal{P}}(\cdot) \) and \( F_{\mathcal{Q}}(\cdot) \) are the probability functions. \( F_{\mathcal{Q}}(\cdot) \) describes the probability that a \( x \in \mathbb{R} \) with a given distribution over \( \mathcal{Q} \) will be found to have a value which satisfies the condition within the parentheses. Function \( F_{\mathcal{P}}(\cdot) \) holds the same definition. Function \( f_{\mathcal{P}}(p) \) represents probability density function which equals to \( 1/\|\mathcal{P}\| \), \( f_{\mathcal{Q}}(q) \) equals to \( 1/\|\mathcal{Q}\| \). On the other hand, we define the expectation considering local rigidity as \( E_2 \):

\[
E_2 := \int_{-\infty}^{\infty} \left( f_{\mathcal{P}}(p)F_{\mathcal{Q}}(|x - p|_2 \leq |p - q|_2)\right)^{-1} \times f_{\mathcal{Q}}(q)F_{\mathcal{P}}(|x - q|_2 \leq |p - q|_2)\|\mathcal{P}\|^{-1} dp dq, \tag{8}
\]

where \( x = \{x^-, x^+\}, p = \{p^-, p^+\}, q = \{q^-, q^+\} \). Variable \( p^- \in \mathcal{P} \) and \( p^+ \in \mathcal{P} \) are the the closest left point and the closest right point to \( p \in \mathcal{P} \) respectively. The meaning of this denotation also applies to \( x \) and \( q \). From Fig 6, we can observe two properties, 1) higher expectation can be observed when parameters \( \mu, \sigma \) are closer to \((0, 0.1)\); 2) in (b), the expectation drops faster than (a) when \((\mu, \sigma)\) become larger than \((0, 0.1)\). These two figures show that our method is more sensitive with the difference of distribution and thus results in better performance. These two properties we observed can well prove the two assumptions we made.

4. Experiment

4.1 Experiment Environment

We use the benchmark\(^1\) used in [4] to evaluate our method. This benchmark is inherited from online tracking benchmark [21]. Hence, it is very challenging for global template matching task. Many real-world difficulties have been considered in this benchmark such as occlusion, illumination change, background clutter, deformation, etc. There are 106 pairs of template and target images in this benchmark with various image size. All the ground truth bounding boxes are annotated manually with a semantic foreground defined.

We use the overlap rate to judge whether a matching result is successful by referring to the ground truth. Specifically, PASCAL criteria [22] is used to calculate the overlap rate:

\[
\text{overlap rate} = \frac{\text{area}(BB_{re} \cap BB_{gr})}{\text{area}(BB_{re} \cup BB_{gr})}. \tag{9}
\]

Where \( BB_{re} \) means bounding box of result and \( BB_{gr} \) means bounding box of ground truth. \( \text{area}(\cdot) \) is a function to count number of pixels. Based on the overlap rate, we can achieve the answer about whether a matching result is correct or wrong by setting a threshold. Specifically, we have

\[
\text{answer} = \begin{cases} 
1 & \text{if overlap rate} > \text{threshold} \\
0 & \text{otherwise}
\end{cases}. \tag{10}
\]

Finally, success ratio = \#(answer|answer = 1)/#test as the

---

\(^1\)Full results of this benchmark can be viewed at [http://cvhost.scv.cis.iwate-u.ac.jp/research/projects/nonparametric_template_matching.html](http://cvhost.scv.cis.iwate-u.ac.jp/research/projects/nonparametric_template_matching.html). For better repeatability, source code will be publicly available after any possible acceptance.
Fig. 5 2D case with synthetic data generated by Gaussian distribution. a) We first generate 100 blue points from a normal distribution $N(\mu_1, \sigma_1)$. To give each blue point rigidity, we generate four points around each blue point vertically and horizontally. Totally, the combination of 500 blue points is treated as foreground model. As shown in the enlarged part of (a), the combination of blue points shows like a cross. We then generate 500 red points as background from a different distribution $N(\mu_2, \sigma_2)$. b) Similarly, we first generate 100 blue points from $N(\mu_1, \sigma_1)$ and then extend them to 500. Background is drawn from $N(\mu_3, \sigma_3)$. c) The matching result of (a) without considering the LRC. d) The matching result of (b) without considering the LRC. e) The matching result of (a) considering LRC. f) The matching result of (b) considering LRC.

accuracy criterion.

All the experiments have been done on a PC equipped with Intel Core-i7 2.9GHz and 16 GB RAM.

4.2 Comparison

We compare our method with both classical methods and state-of-the-art methods. Classical methods such as SAD, SSD, HM and NCC have been comprehensively studied in [23]. Among recent methods, BB, BDS are patch-to-patch similarity measurements, which are closest to our method. HOG is a dense feature combined with SSD during the comparison. Figure 8(a) illustrates the comparison result of accuracy at a glance. For clarity, we dynamically change the threshold and each threshold corresponds to a success ratio value. Each curve represents one method’s result and it is worth noting that BB only partially improves the accuracy against previous methods when the threshold is smaller than 0.63. When the threshold exceeds 0.63, other methods such as HOG, SAD can even outperform BB. This is because dense feature matching methods can adjust the location of final matching result better when less deformation occur. On the other hand, LRC can not only improve the success ratio in case of threshold < 0.63, but also maintain the same level of accuracy with dense feature matching method when the overlap rate becomes higher. Table 1 shows the average success ratio over all the matching tests in the benchmark. BB improves the accuracy by 5% and LRC improves the accuracy against BB by 4%.
Fig. 8 Comparative result. a) Success ratio curves with threshold of overlap rate be changed from 0 to 1. b) Case-by-case comparison with BB. LRC improves the overlap rate on many test cases in the benchmark.

Fig. 9 Effect of parameters on success ratio. (a) Varying the parameter of location feature’s weight $w_l$ from 0.5 to 3.0. (b) Varying the parameter of color feature’s weight $w_c$ from 0.5 to 3.0. (c) Varying the parameter of spatial feature’s weight $w_g$ from 0.5 to 3.0. (d) Varying the parameter $\sigma$ which affects the degree of smoothness from 0.1 to 0.9. (e) Comparing the results over three different color spaces. (f) Varying the parameter of patch size $s$ from 3 to 6.

4.3 Effect of Parameters

In this section, we systematically report the results for studying how each parameter affects the performance of our matching method. Six parameters $w_l$, $w_c$, $w_g$, $\gamma$, $C$, $s$ are studied which have been mentioned in Sect. 3. The results are concluded in Fig. 9. From (a) to (f) we can see that all the six parameters affect the final result in a certain extent. The best performance is achieved when $w_l = 2$, $w_c = 1$, $w_g = 1$, $\gamma = 0.6$, $C = RGB$, $s = 3$. All the solid curves show the parameters we have used in the comparative experiment.
Unexpectedly, increasing the patch size will cause a sharp decrease on accuracy, that means our method needs to pay a certain amount of computational cost to keep the accuracy. In our implement, about 2 seconds are needed for matching a 480×270 pixel target image with 19×45 pixel template. Processing time is directly proportional to the template size and target size. In addition to the patch size, smooth level also affect the performance a lot. Smoothness assumption is a very important precondition for template matching. An edge image (which is not smooth) without preprocessing is not suitable for template matching since a little displacement will change the matching score drastically. Over-smoothed images will also lose important feature information and lead to failed matching. Figure 10 shows some examples of matching results with tuned parameters. Our approach succeeded in many different conditions such as: drastic appearance change, illumination change, small size, etc.

5. Conclusion and Future Work

In conclusion, this paper presents a template matching method which has no need to define a specific deformation model. Local rigidity constraint (LRC) has been proposed, which is defined as a pair of matched patches. Counting number of LRC is treated as the visual similarity between a template and a candidate. All the one-dimensional, two-dimensional synthetic experiments and real matching test show the efficiency of considering the local rigidity (if any) can improve the matching accuracy. However, several drawbacks have limited the application of this method. 1) Since only translation has been considered, scaling and rotation cannot be sensed during the matching. 2) The matching accuracy of non-rigid objects, such as fluid, can hardly be improved.

As the future work, we intend to enhance this method for more intense environment changes in order to solve the problems which have been reflected in most of the failure tests. Finally, we hope to improve the matching accuracy and expect further real-world applications.

References

1981.


Chao Zhang received a B.E. in the College of Information Engineering from Northwest A & F University in 2011. He is now a Ph.D. candidate at Design and Media Technology, Graduate School of Engineering, Iwate University and has received a M.E. His research interests include image matching, online tracking, and object detection. He is a student member of the IEICE and IEEE.

Haitian Sun received a B.E. in the College of Information Engineering from Northwest A & F University in 2014. He is now a Ph.D. candidate at Design and Media Technology, Graduate School of Engineering, Iwate University. His research interests include sketch based image retrieval, human pose estimation. He is a student member of the IEEE.

Takuya Akashi received a Ph.D. in the System Design Engineering from the University of Tokushima in 2006. Since April 2009, he has been with the Department of Electrical Engineering, Electronics and Computer Science, Iwate University. In 2015, he is a visiting associate with the California Institute of Technology. He is currently an associate professor with the Iwate University. His research interests include evolutionary algorithms, image processing and human sensing. He is a member of the IEEE.