Research Article

A Geometrical-Information-Assisted Approach for Local Feature Matching

Buhai Shi, Qingming Zhang, and Haibo Xu

School of Automation Science and Engineering, South China University of Technology, Guangzhou, Guangdong, 510641, China

Correspondence should be addressed to Qingming Zhang; autox2015@mail.scut.edu.cn

Received 5 August 2018; Revised 20 January 2019; Accepted 5 February 2019; Published 24 February 2019

Academic Editor: Paolo Spagnolo

Copyright © 2019 Buhai Shi et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper presents a geometrical-information-assisted approach for matching local features. With the aid of Bayes’ theorem, it is found that the posterior confidence of matched features can be improved by introducing global geometrical information given by distances between feature points. Based on this result, we work out an approach to obtain the geometrical information and apply it to assist matching features. The pivotal techniques in this paper include (1) exploiting elliptic parameters of feature descriptors to estimate transformations that map feature points in images to points in an assumed plane; (2) projecting feature points to the assumed plane and finding a reliable referential point in it; (3) computing differences of the distances between the projected points and the referential point. Our new approach employs these differences to assist matching features, reaching better performance than the nearest neighbor-based approach in precision versus the number of matched features.

1. Introduction

For matching local features, the threshold-based method and the nearest neighbor-based approach (NNA) are two fundamental strategies. Compared with the threshold-based approach, the NNA matches features more precisely [1]. From the previous works (cf. [2–5]) to the recent ones (cf. [6, 7]), the nearest neighbor techniques are exploited broadly. Since positions of matched features can be seen as samples generated from two images related by a certain homography, RANSAC [8] is usually applied to exclude the impact of outliers on the estimation [9–16], therefore improving matching effect of local features. RANSAC has a defect when inliers are fairly less in putative matches, which causes unpredictable elapsed time, even failing to estimate consensus sets. For matching features more efficiently, some methods introduce prior information into the algorithm of estimating consensus sets, e.g., Guided-MLESAC [17], PROSAC [18], and SESAC [19], using distributions constructed by prior information instead of the uniform distribution to generate hypothetical homographies. Besides, some preliminary processes are also employed to produce subsets containing more uncontaminated samples in advance before running plain RANSAC. For example, Cov-RANSAC [20] employs SPRT and a covariance test; DT-RANSAC [21] refines data by topological information in the Delaunay triangulation; SVH-RANSAC [22] adopts a local feature scale constraint to group observations; SC-RANSAC [23] utilizes spatial relations between extracted corresponding points; and WD-RANSAC [24] and QML-RANSAC [25] adopt the Wigner distribution and the quasi maximum likelihood algorithm, respectively. Nevertheless, in the case of matching features between nonrigidly transformed images, it is difficult to estimate a nonparametric consensus by a homography-estimation-based approach. Some nonparametric consensus methods are also developed to match local features. Sparse-VFC [26, 27] introduces sparse representations to estimating vector field consensus and shows powerful performance in matching features. LLT [28] exploits local geometrical constraints to estimate the consensus set which comprises inliers in matches between two rigidly or nonrigidly transformed images. Besides those consensus-estimation-based methods, approaches without estimating consensus are also studied. The method presented in [29] divides the vector of a local descriptor into subvectors and employs binary trees to compare subvectors to achieve NNA. For matching features in the binocular stereo scene, the approach in [30] uses some adjacent pixels near a feature point to form a block and...
then to find the best match by the block. LPM [31] applies local neighborhood structures to determine true matches, and based on it, GLPM [32] introduces a set with more confidence of including true matches, which are obtained by the distance ratios of local descriptors, and then reaches better performance than LPM. Another local-geometrical-information-based technique is proposed by [33] for describing features and matching features, which exploits topological relationship amid local features. Moreover, in the case of matching some deep features, convolutional neural networks are employed [34].

Motivated by approaches above built on local geometrical information, we discuss a new method assisted by geometrical information to improve the performance for matching features. An extracting procedure for scale-invariant features provides information about scale or even affine parameters of features (cf. [10, 35]), which can estimate the geometrical relationship between two images. We here discover an appropriate geometrical relation and study how to estimate it and exploit this relation to further improve matching effect. This paper is organized as follows. In Section 2, we discuss factors that influence matching effect and point out a new way to enhance matching effect. In Section 3, we discuss how Euclidean distance can assist matching features and work out an algorithm of matching features assisted by geometrical information. We would like to abbreviate the algorithm Geometrical-Information-Assisted Matching as GIAM in this paper. In Section 4, we test GIAM and compare the results with other methods. Finally we conclude our work in Section 5.

2. Factors That Influence Matching Effect

In what follows, Pair(P, Q) denotes the event that a point P matches another point Q, and ~Pair(P, Q) is the negation of it. F(⋅) denotes a feature descriptor and d(⋅, ⋅) is a similar metric of the descriptors of two points. Besides, the symbol c(ε) represents that the function c(ε) → e (ε is a constant satisfying 0 ≤ e ≤ 1) as ||ε|| → 0, where e is a vector in Rn for some positive integer n.

2.1. Discussion for Posterior Confidence of Matched Features.

Usually it is not difficult to construct a descriptor F(⋅) and a similar metric d(⋅, ⋅) satisfying

\[ Pr\left(d\left(F(P), F(Q)\right) ≤ ε \mid Pair(P, Q)\right) = 1(e) \]

where P and Q are two points of attempt matching. Suppose Pr(Pair(P, Q)) = p. For a certain descriptor and a similar metric, we have

\[ Pr\left(Pair(P, Q) \mid d\left(F(P), F(Q)\right) ≤ ε\right) = 1(e)p \]

Hence the posterior probability of the matched P and Q is

\[ Pr\left(Pair(P, Q) \mid d\left(F(P), F(Q)\right) ≤ ε\right) = \frac{1(e)p}{1(e)p + R} \]

where R = Pr(d(F(P), F(Q)) ≤ ε | ~Pair(P, Q))(1 − p).

As can be seen from (3), except the trivial case of p = 1, to reach a higher posterior confidence of the match (P, Q), it is plausible to lower the conditional probability Pr(d(F(P), F(Q)) ≤ ε | ~Pair(P, Q)). Nevertheless, comparing with designing an appropriate F(⋅) and d(⋅, ⋅) satisfying

\[ Pr\left(d(F(P), F(Q)) ≤ ε \mid Pair(P, Q)\right) = 1(e), \text{ constructing F(⋅) and d(⋅, ⋅) such that Pr}\left(d(F(P), F(Q)) ≤ ε \mid ~Pair(P, Q)\right) = 0(e) \]

is a more arduous task. Thus it would be alleviating if another metric m(⋅, ⋅) can be introduced satisfying

\[ Pr\left(m(P, Q) ≤ ε \mid ~Pair(P, Q)\right) = e(ε) \]

for a given ~Pair(P, Q) and for some 0 < e ≤ 1, where the event \{m(P, Q) ≤ ε | ~Pair(P, Q)\} is conditionally independent with \{|d(F(P), F(Q)) ≤ ε| ~Pair(P, Q)\}. Let us denote \( Pr(d(F(P), F(Q)) ≤ ε) \equiv c_1(e) \) and \( Pr(d(F(P), F(Q)) ≤ ε)) \equiv c_2(e, ε) \), so that the posterior confidence of matched features can be written as

\[
Pr\left(Pair(P, Q) \mid d(F(P), F(Q)) ≤ ε\right) = \frac{c_2(e_1, ε_2)p}{c_2(e_1, ε_2)p + (c_1(e_1) - 1(e_1)p) \cdot e(ε)}
\]

In contrast to (3), (5) shows that e(ε) improving posterior confidence. As will be shown next, some geometrical information can be exploited to construct a metric satisfying (4).

2.2. Geometrical Information for Matching Features. Suppose that two images I_p and I_Q are transformed, respectively, from an image I_com by two affine homographies

\[
H_p = \begin{pmatrix} T_p & t_p \\ 0 & 1 \end{pmatrix},
\]

\[
H_Q = \begin{pmatrix} T_Q & t_Q \\ 0 & 1 \end{pmatrix},
\]

where t_p and t_Q contribute to translations on the homogeneous coordinates; T_p and T_Q are invertible linear transformations on the Cartesian coordinates, which yield rotation, scale change, and tilt of planar images. Let symbols P_p and Q_Q be matched points and symbols P and Q be points (respectively, in I_p and I_Q) to be matched. Here we by “a point in an image”, say P in I_p, mean that P ∈ dom(I_p), its domain of definition. Since the procedure for feature extracting hardly provides information about t_p and t_Q, we then perform a technique to erase them by calculation of Euclidean distance (which is denoted by d_ε(⋅, ⋅)) in the same image I_com as follows:

\[
d_\varepsilon\left(T_p^{-1}(P-t_p), T_p^{-1}(P_o-t_p)\right) = d_\varepsilon\left(T_r^{-1}P, T_r^{-1}P_o\right),
\]

and

\[
d_\varepsilon\left(T_Q^{-1}(Q-t_Q), T_Q^{-1}(Q_o-t_Q)\right) = d_\varepsilon\left(T_Q^{-1}Q, T_Q^{-1}Q_o\right).
\]

Therefore we figure out the following expression describing geometrical information (where i = 1, ..., n)

\[
GI(P, Q) = d_\varepsilon\left(d_\varepsilon\left(T_p^{-1}P, T_p^{-1}P_o\right), d_\varepsilon\left(T_Q^{-1}Q, T_Q^{-1}Q_o\right)\right).
\]
Noting that the point \( T_p^{-1}(P_0 - t_p) \) equals \( T_Q^{-1}(Q_0 - t_Q) \) in the image \( I_{com} \), we will show that the geometrical information \((9)\) can be employed as a metric satisfying \((4)\) and \((5)\). We call such a \( T_Q^{-1}(P_0 - t_p) \) the referential point and \( (P_0, Q_0) \) the referential pair.

**Proposition 1.** Suppose that the image \( I_{com} \) is defined on \( \mathbb{R}^2 \). Then any point \( P \) in \( I_p \) and any point \( Q \) in \( I_Q \) satisfy

\[
\Pr\{GI(P, Q) \leq \varepsilon \mid \text{Pair}(P, Q)\} = 0 \quad (\varepsilon) .
\]

**Proof.** Denote by \( p(x, y) \) the p.d.f. of projected feature points \((x, y)\) in \( I_{com} \) (from \( \text{dom}(I_p) \) or \( \text{dom}(I_Q) \)) on \( \mathbb{R}^2 \) and set \( r = d_E(T_p^{-1}P, T_p^{-1}P_0) \). Since \( P \) and \( Q \) do not match, the point \( T_Q^{-1}(P - t_p) \) differs from \( T_Q^{-1}(Q - t_Q) \) in \( \mathbb{R}^2 \). We then have

\[
\Pr\{GI(P, Q) \leq \varepsilon \mid \text{Pair}(P, Q)\} = \int_{D_1} p(x, y) dx dy,
\]

where \( D_1 = \{(x, y) \mid (r - \varepsilon)^2 \leq x^2 + y^2 \leq (r + \varepsilon)^2\}. \) It follows that \( \Pr\{GI(P, Q) \leq \varepsilon \mid \text{Pair}(P, Q)\} \to 0 \) as \( \varepsilon \to 0 \). \( \Box \)

The result of Proposition 1 indicates an optimal case in applying \((5)\). However, we need to formulate the discrete form for pragmatic use, which is built on a probability dominated by the counting measure.

**Proposition 2.** Suppose that \( I_{com} = f(x_k, y_l) \) is an image in its discrete form, where \( k = 1, 2, \ldots, M, l = 1, 2, \ldots, N \) for some \( M, N \in \mathbb{Z}_+ \). Then any point \( P \) in \( I_p \) and any point \( Q \) in \( I_Q \) satisfy

\[
\Pr\{GI(P, Q) \leq \varepsilon \mid \text{Pair}(P, Q)\} = c(\varepsilon),
\]

where \( 0 \leq c < 1 \).

**Proof.** Denote by \( p_{kl} \) the probability of a feature point (from \( \text{dom}(I_p) \) or \( \text{dom}(I_Q) \)) being projected to the point \((x_k, y_l)\) and set \( r = d_E(T_p^{-1}P, T_p^{-1}P_0) \). Since \( P \) and \( Q \) do not match, the point \( T_Q^{-1}(P - t_p) \) differs from \( T_Q^{-1}(Q - t_Q) \) in the image \( I_{com} \). Hence we have

\[
\Pr\{GI(P, Q) \leq \varepsilon \mid \text{Pair}(P, Q)\} = \sum_{(x_k, y_l) \in D_2} p_{kl} < 1 ,
\]

where \( D_2 = \{(x, y) \in \text{dom}(I_{com}) \mid (r - \varepsilon)^2 \leq x^2 + y^2 \leq (r + \varepsilon)^2\} \). We finish the proof by noting that for sufficiently small \( \varepsilon \)'s, the \( D_2 \)'s are constant. \( \Box \)

Since the number of pixels centred at the referential point with a certain radius is far less than the number of pixels in the whole image, presumably in most cases the \( c \) should be far less than 1. Therefore Proposition 2 offers a suboptimal result for \((5)\) in matching features. The proofs and discussions of Propositions 1 and 2 are shown in Figure 1.

### 3. Matching Features Assisted by Geometrical Information

#### 3.1. Discussion for Algorithm

We first discuss how to obtain \( T_p^{-1} \) and \( T_Q^{-1} \). Suppose that there is a linear transformation \( T \) mapping an unit circle \( \{x^2 + x^2 = 1\} \), to an ellipse \( \{ay^2 + 2by_1y_2 + cy_2^2 = 1\} \), and denote \( A = (a, b, c) \), \( x = (x_1, x_2)^T \), and \( y = (y_1, y_2)^T \). Then we have

\[
\begin{align*}
& y = Tx \\
& x^T x = 1 \\
& (T^{-1} y)^T (T^{-1} y) = 1 \\
& y^T (T^{-1})^T (T^{-1}) y = 1 \\
& A = (T^{-1})^T (T^{-1}),
\end{align*}
\]

which means that by Cholesky decomposition of the positive-definite matrix \( A \), we can obtain the linear transformation \( T^{-1} \). In descriptors of Mikolajczyk's format \([10, 35]\), the affine region of a feature point is enclosed by an ellipse with parameters \(a, b, c\) (for the details of these parameters please refer to the website http://www.robots.ox.ac.uk/~vgg/research/affine/descriptors.html#binaries), by which we can calculate \( T_p^{-1} \) and \( T_Q^{-1} \).

Next we need to find the referential pair \((P_0, Q_0)\) and to estimate the invertible linear transformations \( T_p \) and \( T_Q \). Suppose that \( \{(P_i, Q_i)\}_{i=1}^N \) is a sequence, which consists of the pairs ordered in similar metrics of descriptors. We assign the most similar pair as the referential pair, i.e., \((P_0, Q_0) = (P_I, Q_I)\). Since the unit circle in \( I_{com} \) can be transformed by \( T_p \) and \( T_Q \) onto the boundaries (ellipses) of affine regions of feature points \( P_i \) and \( Q_i \), respectively, \( T_p^{-1} \) and \( T_Q^{-1} \) should be the respective Cholesky decomposition of matrices \((a_{ii}, b_{ii}, c_{ii})\) and \((a_{jj}, b_{jj}, c_{jj})\), where \( a_{ii}, b_{ii}, c_{ii} \) are parameters of the boundary ellipse of the affine region of \( P_i \) and the same applies to \( a_{jj}, b_{jj}, c_{jj} \).

The geometrical information for each pair of feature points can be calculated henceforth. For two sets of feature points \( \{P_i\}_{i=1}^M \) and \( \{Q_i\}_{i=1}^N \), we compute their geometrical information as

\[
GI(P_i, Q_j) = d_E(T_p^{-1}P_i, T_p^{-1}P_0, d_E(T_Q^{-1}Q_j, T_Q^{-1}Q_0)).
\]

Finally, new matches are chosen by ordered scores which are calculated by a sum of similar metrics of features and geometrical information of feature points. We calculate the score of each putative pair as

\[
score(P_i, Q_j) = d(F(P_i), F(Q_j)) + GI(P_i, Q_j).
\]

#### 3.2. Data Objects and the Algorithm of GIAM

According to the preceding discussion, we assign five matrices, including a
geometrical information of feature points, which is given by

\[\text{projected into the same circle centred at the referential point is zero under the Lebesgue measure on } \mathbb{R}^2, \text{ whereas the right side shows that the counterpart may not be zero under the counting measure since there are points except } T_{pq}^{-1}(P - tp) \text{ (e.g., the 11 pink boxes above) that } Q \text{ could be projected to.}\]

![Figure 1: An illustration of $I_{com}$ for Propositions 1 and 2, where the left side indicates that the probability of two points $P$ and $Q$ being projected into the same circle centred at the referential point is zero under the Lebesgue measure on $\mathbb{R}^2$, whereas the right side shows that the counterpart may not be zero under the counting measure since there are points except $T_{pq}^{-1}(P - tp)$ (e.g., the 11 pink boxes above) that $Q$ could be projected to.](image)

**Algorithm 1: Algorithm of GIAM (Geometrical-Information-Assisted Matching).**

\[
\text{Input: Two sets of descriptors in Mikolajczyk's format, } \\
\Omega_p = \{P_i = (x_i, y_i), a_i, b_i, c_i (codes)\}_{i=1}^M \text{ and } \Omega_Q = \{Q_j = (x_j, y_j), a_j, b_j, c_j (codes)\}_{j=1}^N. \\
\text{(a) Calculate the descriptor difference matrix } D. \\
\text{(b) Pick out the most similar pair of feature points by using the descriptor difference matrix } D \text{ and assign it as the referential pair } (P_o, Q_o). \\
\text{(c) Estimate the } T_{pq}^{-1} \text{ and } T_{pq}^{-1} \text{ in Eq. (15).} \\
\text{(d) Calculate the distance correlation matrix } C_i \text{ by Eq. (15) and (17). Since there is only one referential pair, the order of the distance correlation matrix is 1; } \\
\text{(e) Calculate the score matrix } S_i = D + C_i. \\
\text{(f) Compute an ordered matched list } \{(P_o, Q_o)\}_{k=1}^L \text{ by using the score matrix } S_1, \text{ where } \\
L = \min\{M, N\} \text{ is the number of matched feature pairs.} \\
\text{Output: An ordered matched list } \{(P_o, Q_o)\}_{k=1}^L.
\]

**4. Simulations**

4.1. **Datasets and Methods for Simulations.** To test our new algorithm for local feature matching, we employ four methods, NNA, NNA with the plain RANSAC (NNA-RANSAC), NNA with LPM (NNA-LPM), and NNA with...
GIAM (NNA-GIAM). The codes of RANSAC adopted in our simulations are developed by Marco Zuliani (all these codes are obtained from the website https://github.com/RANSAC/RANSAC-Toolbox) and the codes of LPM are developed by Dr. Jiayi Ma (all these codes are obtained from the website https://github.com/jiayi-ma/VFC). We set parameters for LPM as shown in Table 1 and parameters for RANSAC as shown in Table 2.

We utilize an executable file implemented by Dr. Mikolajczyk (the original codes are obtained from the website http://www.robots.ox.ac.uk/~vgg/research/affine/) to produce experimental data. First, we use the executable file to process all test images and generate the sets of descriptors for each test image. Second, we match features between those sets by NNA, NNA-RANSAC, and NNA-LPM as well as NNA-GIAM, respectively. We here define that a point in the first image correctly matches a corresponding point in the second image, if the distance between the point in the second image and the standard point (to which the point in the first image is mapped via a given homography which is a ground truth) is less than 2 pixels. Precision of matches with regard to a number of matched features equals the number of correct matches divided by the number of matches. We apply a curve of precision vs. number of matches to estimate the performance of the four algorithms.

For tests under fairly complex scenes, such as changes from slight to dramatic degree of rotation, scale, viewpoint, blur, light, and JPEG compression, the Mikolajczyk's test data [36] are exploited, which consist of a group of image sets with homographies as the ground truth (the image sequences are from the website http://www.robots.ox.ac.uk/~vgg/research/affine/detectors.html) There are 8 test sets in the group, and each set contains 6 images. We match features in the first image to the rest of images in each test sequences by NNA, NNA-RANSAC, and NNA-LPM as well as NNA-GIAM, respectively. In these tests we employ the Harris-Affine detector [10, 35] to extract features and describe them by the SIFT [3, 4].

### Table 1: Parameters in LPM.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\lambda_1$</td>
<td>0.8; $\lambda_3$</td>
</tr>
<tr>
<td>The number of nearest neighbour 1</td>
<td>6;</td>
</tr>
<tr>
<td>The number of nearest neighbour 2</td>
<td>6;</td>
</tr>
<tr>
<td>$r_1$</td>
<td>0.2; $r_3$</td>
</tr>
</tbody>
</table>

### Table 2: Main parameters in RANSAC.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chi-squared probability threshold for inliers</td>
<td>$1 - e^{-4}$</td>
</tr>
<tr>
<td>False alarm rate</td>
<td>$e^{-6}$</td>
</tr>
<tr>
<td>Algorithm flavour</td>
<td>RANSAC</td>
</tr>
<tr>
<td>Maximum number of iterations</td>
<td>100000</td>
</tr>
<tr>
<td>Minimum number of iterations</td>
<td>1</td>
</tr>
<tr>
<td>The $\sigma$ of assumed Gaussian noise</td>
<td>1</td>
</tr>
</tbody>
</table>

In these figures, the “1vs.k” in the captions is interpreted as “the first image versus the k-th image”, indicating that the test result is obtained by matching features between the first image and the k-th image in the same test image sequence.

In the case of rotation and scale change for the textured scene (cf. Figure 2), GIAM overcomes NNA and exceeds LPM and RANSAC when the change is dramatic. After the middle degree change ((c), (d), (e) in Figure 2), GIAM outperforms LPM on most intervals of “number of matched features”. GIAM shows slight advantage to NNA on the rotation and scale change for the structured scene (cf. Figure 3). In the cases of blurred images on both structured scene and textured scene (cf. Figures 4 and 5), GIAM improves precision of NNA. GIAM reaches higher scores than LPM when the number of matches is relatively small ((a), (b), (c) in Figure 4) in the structured scene test, and admits analogous precision to RANSAC. For the severest JPEG compression, GIAM has better performance than NNA and LPM (cf. Figure 6). GIAM, NNA, and RANSAC show almost identical performance in the case of slight to mild JPEG compression over more than half of the matches. LPM finds relatively fewer correct matches in this test sequence. In the case of illumination change (cf. Figure 7), GIAM reaches higher precision than NNA as the light turns dim and in general slightly outperforms LPM over the first 1/3 of all matches in each degree of illumination. In the case of viewpoint change for the textured scene (cf. Figure 8), GIAM overcomes LPM and exceeds NNA in all tests, and has comparable performance with RANSAC over the first 1/2 of all matches. Under the situation of viewpoint change for the structured scene (cf. Figure 9), GIAM exceeds NNA in all tests and shows better performance than others when the number of matches and the change of viewpoints are relatively small ((a), (b) in Figure 9).

Consequently, it can be seen from these results that GIAM improves precision of NNA on the situations including scale change, blur change, JPEG compression change, illumination change, and slight change in viewpoint, and reaches better performance than LPM in some change of rotation, scale, illumination, signal compression, and viewpoint.

### 5. Conclusion

We have studied how geometrical information improves the posterior confidence of matched features and therefore enhances the matching effect. There are two essential merits in our work. The first one is that we utilize Bayes’ theorem to analyze factors influencing matching effect and figure out that prior information, which is a conditional distribution satisfying (4), can assist improving the posterior confidence of matches. The second one is exploiting the geometrical information in the descriptors, which can be used to estimate the distance between projected feature points. Consequently, the technique proposed in this paper shows its capability to improve the performance of matching features.
Figure 2: Rotation and scale change for the textured scene by the Bark sequence, where (a), (b), (c), (d), (e) show test scores for 1vs.2, 1vs.3, 1vs.4, 1vs.5, respectively.
Figure 3: Rotation and scale change for the structured scene by the Boat sequence, where (a), (b), (c), (d), (e) show test scores for 1vs.2, 1vs.3, 1vs.4, 1vs.5, respectively.
Figure 4: Blur for the structured scene by the Bikes sequence, where (a), (b), (c), (d), (e) show test scores for 1vs.2, 1vs.3, 1vs.4, 1vs.5, respectively.
Figure 5: Blur for the textured scene by the Trees sequence, where (a), (b), (c), (d), (e) show test scores for lvs.2, lvs.3, lvs.4, lvs.5, respectively.
Figure 6: JPEG compression by the UBC sequence, where (a), (b), (c), (d), (e) show test scores for lvs.2, lvs.3, lvs.4, lvs.5, respectively.
Figure 7: Illumination change by the Leuven sequence, where (a), (b), (c), (d), (e) show test scores for 1vs.2, 1vs.3, 1vs.4, 1vs.5, respectively.
Figure 8: Viewpoint change for the textured scene by the Wall sequence, where (a), (b), (c), (d), (e) show test scores for lvs.2, lvs.3, lvs.4, lvs.5, respectively.
Figure 9: Viewpoint change for the structured scene by the Graffiti sequence, where (a), (b), (c), (d), (e) show test scores for 1vs.2, 1vs.3, 1vs.4, 1vs.5, respectively.
Appendix

Results of Simulations

See Figures 2–9.

Data Availability

The image and diagram data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work is supported by Guangdong Project of Science and Technology Development (2014B09091042) and Guangzhou Sci & Tech Innovation Committee (201707010068). The authors appreciate Dr. Krystian Mikolajczyk for his test data and an executable file, Dr. Macro Zuliani for his codes of RANSAC, and Dr. Jayi Ma for his codes of LPM.

References


