Two-Frame Wide Baseline Matching

Jiangjian Xiao Mubarak Shah

Computer Vision Lab, School of Electrical Engineering and Computer Science
University of Central Florida, Orlando, Florida 32816, USA
{jxiao, shah}@cs.ucf.edu

Abstract

This paper describes a novel approach to automatically recover corresponding feature points and epipolar geometry over two wide baseline frames. Our contributions consist of several aspects: First, the use of an affine invariant feature, edge-corner, is introduced to provide a robust and consistent matching primitives. Second, based on SVD decomposition of affine matrix, the affine matching space between two corners can be approximately divided into two independent spaces by rotation angle and scaling factor. Employing this property, a two-stage affine matching algorithm is designed to obtain robust matches over two frames. Third, using the epipolar geometry estimated by these matches, more corresponding feature points are determined. Based on these robust correspondences, the fundamental matrix is refined, and a series of virtual views of the scene are synthesized. Finally, several experiments are presented to illustrate that a number of robust correspondences can be stably determined for two wide baseline images under significant camera motions with illumination changes, occlusions, and self-similarities. After testing a number of examples and comparing with the existing methods, the experimental results strongly demonstrate that our matching method outperforms the state-of-art algorithms for all of the test cases.

1. Introduction

Tracking feature points is a classic and the basic computer vision problem. Currently, the tracking technique over *small* baseline, such as a video sequence, is fully developed [11, 3]. However, a number of very interesting applications need to obtain reliable corresponding features over a *wide* baseline, such as view morphing [10] and reconstruction from multiple images [2, 4, 9]. In these applications, the external and internal parameters between any two views are significantly different, and illumination may also be significantly different. Therefore, the features corresponding over two wide baseline frames cannot be effectively determined via the traditional matching or tracking methods due to the large geometric transformation and illumination changes. Fig.1 shows two sequences of wide baseline frames at different viewing angles.

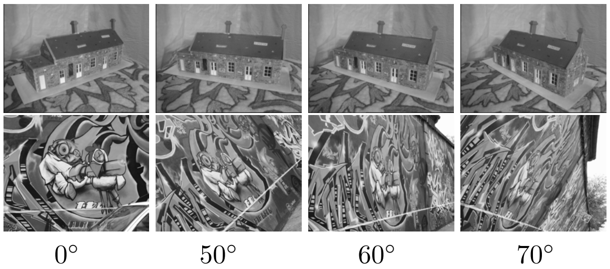


Figure 1: Wide baseline frames of “House” sequence from Oxford Univ. are 3D views, and “Graffiti-6” sequence from INRIA are planar views (Homography). The corresponding viewing angle is indicated below each image.

In this paper, we present a new approach to obtain a number of reliable corresponding points. First, an edge-corner detector is used to determine affine invariant edge-corners, which guarantee that the corners are located at the intersection of two or more edges. Second, after decomposing the affine matrix, we found that the affine matching space between two corners can be approximately divided into two independent spaces by rotation angle and scaling factor. Therefore, a two-stage affine matching algorithm is designed to obtain robust matches over these two frames to recover geometric transformation and illumination changes between corresponding points. In the first stage, we explicitly search these two spaces to obtain a good initial state for the gradient descent minimization used in the second stage. Third, we use the robust fundamental matrix estimation [4, 16, 2] to eliminate the outliers and estimate the initial epipolar geometry. Fourth, using the initial epipolar geometry constraint, we recompute the matches in the images by using our two-stage matching algorithm, which can efficiently deal with the self-similarities. Finally, we reestimate the fundamental matrix and get more robust corresponding points, and generate a series of virtual views to synthesize a 3D virtual scene by employing image based rendering technique.

0-7695-1950-4/03 $17.00 © 2003 IEEE

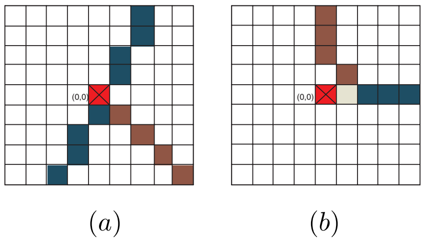


Figure 2: The red pixel '×' is an edge pixel (a) or a pixel neighboring edge pixels (b). It is located at the origin of a $n \times n$ window. The edge passing through the origin can be quickly detected.

2. Related Work

The popular technique for small baseline motion tracking is to minimize the sum of squared differences (SSD) of image intensities over two consecutive frames. However, SSD based frame-to-frame tracker only uses translation motion. Therefore, it may accumulate the errors over time due to other geometric deformation: rotation, and scaling. Shi and Tomasi demonstrated that using an affine model can effectively compensate the errors over video frames [11]. At the same time, they designed a monitor to evaluate the goodness of the correspondences based on the image residuals. Following this direction, Fusiello et al. presented a statistical approach to monitor and reject the outliers [3], and Jin et al. explicitly used illumination information to improve the monitor [5].

Pritchett and Zisserman estimated reliable point correspondences based on local planar homographies [7]. These homographies were determined by parallelogram structures or using motion pyramids. Based on these homographies, the matches can be increased by cross-correlation at the sub-pixel level. However, in general case, the parallelograms that they used may not be present in certain images, or the similar parallelograms may be repeated due to the self-similar structure. Moreover, their method only recovers the homographies related to the parallelograms, and misses some other important homographies implied in the frames.

Baumberg detected Harris features using the second moment matrix and scale space [1]. After obtaining affine texture invariant corner, a Mahalanobis distance metric was used to measure similarity between the two feature vectors. Similarly, Tuytelaars and Van Gool extracted affine invariant regions based on a combination of corners and edges, then matched these features using color moments [14]. Following this direction, Mikolajczyk and Schmid presented the recent work on affine invariant interest point detector [6]. Since they used multi-scale space to determine feature points, their method worked very well for significant scaling case (homography case).

3. Edge-Corners

A corner is a very useful feature for motion correspondence and stereo matching, etc. The common Harris corner detector uses a corner operator,$C = \begin{bmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{bmatrix}$, over a window around a pixel, where C is the second moment matrix. This operator detects point where the image gradients, I_x and I_y, change significantly along the horizontal and vertical directions. Rangarajan et al. defined a corner as the junction of two or more straight line edges, and determined an optimal function to detect the corner[8]. Smith and Brady used non-linear filtering to obtain a fast, noise resistant corner over an image [12]. Recently, Shen and Wang used modified Hough transform to dynamically detect edges and corners [13]. However, their edge detector seems not robust when it is applied to real images. In our experiments, we found that the existing corner detectors usually do not provide consistent and reliable results when the motion gap between two frames is large.

In order to obtain a consistent and affine invariant corner over a wide baseline gap, we combine Canny edge detector, local Hough transform, and Harris corner operator together to design a new matching primitive, edge-corner. One very important property of an edge-corner is that the corner is guaranteed to be located on the intersection of multiple edges. Since the intersection of edges is projected as an edge intersection under the perspective projection (except in the cases when the view point is located on the plane containing these edges), the edge-corner is a consistent, affine invariant primitive for accurate matching over wide baseline frames. Our edge-corner detector is implemented by using the following steps.

First, we use the Canny edge detector to get the edges. Then, we select each edge pixel and its neighboring pixels as corner candidates. Fig.2.b shows that even though the pixel with '×' is not lying at the edge, it still has a chance to become an edge corner.

Next, for each candidate, we consider a $n \times n$ window around this pixel, and set the origin at the window's center to build a local coordinate system (Fig.2). Every straight line across the origin has a fixed tangent value, which can be denoted as $\theta = \tan^{-1}(y/x)$. For each quantized interval θ_i, if the number of edge points is greater than a threshold, there is an edge with angle θ_i passing through the corner candidate. If the number of edges passing this origin is more than 1, this point becomes a potential corner.

Third, we select a threshold λ and use the Harris corner operator to evaluate the goodness of this corner. Then, the non-maxima suppression method is applied to suppress a nearby area to find the best corner candidates. Fig.3 compares Harris corners with edge-corners for "House" frames. Our edge-corners are more precisely located on the junction of edges than Harris corners, and some poor Harris corners

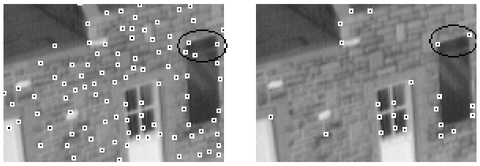


Figure 3: Corners in house frame 0. Left: corners detected by Harris corner detector. Right: edge-corners detected by our method.

are eliminated, which can speed up matching process in the next step.

4. Corner Matching

In this section, we first derive the affine matching equation. Next, we show that using matching scheme with the affine matching equation can obtain stable matches over wide baseline images, which do not contain significant inplane rotation or scaling. Furthermore, we show that using affine matrix decomposition, the affine matching space between two corners can be approximately divided into two independent spaces by rotation angle and scaling factor. Then, a two-stage affine matching algorithm is designed to overcome the scaling and inplane rotation limitations, and to determine reliable matches for general wide baseline images.

4.1 Affine Invariant Matching

Since our edge-corners are affine invariant, the matching between corresponding points can be represented as:

$$I_2(A\mathrm{x} + d) = I_1(\mathrm{x}), \tag{1}$$

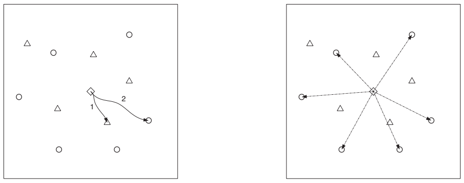
where I_1 and I_2 are two original images, A is a 2D matrix and d is the translation vector. Eq.1 uses an affine motion to compensate the geometric transformation between two image patches (or corners). However, in wide baseline images, the illumination may also change significantly. In order to eliminate the illumination effect, we modify Eq.1 and obtain

$$\mu I_2(A\mathrm{x} + d) + \delta = I_1(\mathrm{x}), \tag{2}$$

where μ depends on the reflection angle of light source, δ depends on the camera gain. We can compute the best match by minimizing the residual

$$\epsilon = \sum_W [(\mu I_2(A\mathrm{x} + d) + \delta) - I_1(\mathrm{x})]^2. \tag{3}$$

This function can be solved by the first order Taylor expansion at $A = \mathrm{I}$ (identity matrix), $d = 0$, $\mu = 1$, and $\delta = 0$.



(a) (b)

Figure 4: Corner matching. Point '$\diamond$' is the starting position of p_{1i}, which is located at the center of a search area (big square). Points '$\triangle$' are the local minima. Points '$\circ$' are the corner points p_{2j}. (a) Search by traditional gradient descent scheme. The search may be stalled at some local minimum such as 1 and 2. (b) corner matching scheme, which directly compares the corners p_{2j} with p_{1i} in the search area.

4.2 Matching Scheme

In wide baseline matching, the search process is more difficult than small baseline frames. Fig.4.a shows a traditional searching scheme, which starts from position of p_{1i} following the initial gradient descent direction. After encountering a local minimum or a corner (corner may be a kind of local minimum), the algorithm will stop and usually can not jump out of the trap by using Newton-Raphson iteration. As a result, the traditional searching scheme usually cannot obtain the correct solution even for simple translation case as shown in Fig.5.a (we used the KLT code of [11], which is available online).

In our scheme (Fig.4.b), we only compare the matches between edge-corners, which can efficiently reduce the search space and avoid the trapping problem due to the local minima. First, we initialize the translation part $d = 0$ in Eq.2 and set the origin of the corner's window at p_{1i} in I_1 and p_{2j} in I_2 respectively. The new equation is represented as $\mathrm{x}_2 = A\mathrm{x}_1 + d$, where x_1 and x_2 are the local coordinates of the corners. Next, we apply this scheme using Eq.2. Since our edge-corner is consistent in most cases, we add a constraint $d \leq 2$ pixels and only allow a small translation of p_{2j}, which can effectively avoid the serious divergence problem in the Newton-Raphson iteration. After the iteration, the corner p_{2j} may move to a new position p'_{2j}, where $d = |p'_{2j} - p_{2j}| \leq 2$. The residue ϵ_{1i2j}(Eq.3) is used to evaluate the goodness of the match. After searching the whole area, we consider the corner p'_{2j} with the smallest ϵ_{1i2j} as the best match for corner p_{1i}. Third, we use the robust fundamental matrix estimation to eliminate the outliers in these matches. In this paper, we use RMS (Root-Mean-Squared distance error (pixel)) to evaluate our results [4]. Fig.5 shows that our method works very well if the major motion of camera is translation (Fig.5.b) or looming ($\leq 2\times$ scaling) (Fig.5.c).

However, in general, the camera motion between the

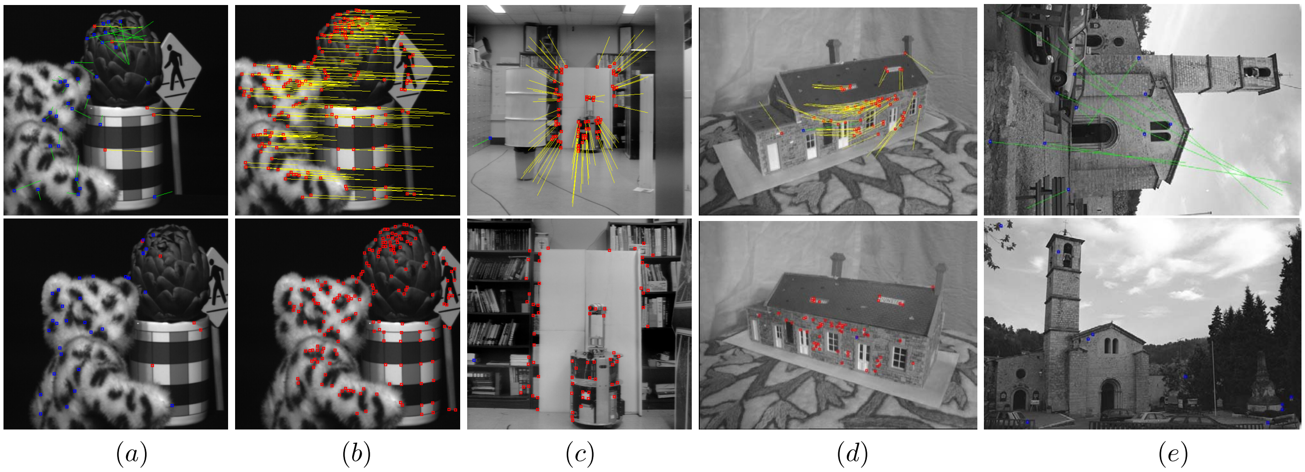


Figure 5: Comparison of KLT algorithm and the corner matching scheme. (a)(frame 0 and 100 of "Artichoke" from CMU) is the result by using KLT code, there are 5 correct matches. $(b)-(e)$ are the results by using the proposed scheme. (b) shows 184 correct inliers for this translation case (a), the RMS is 0.277. (c) (frame 0 and 40 of "Lab" from CMU) shows 67 correct inliers for the looming case, the RMS is 0.435. (d) (frame 0 and 6 of "House") shows 59 correct inliers located on the house, no correct correspondences are recovered on the ground due to the in-plane rotation. (e) (image 3 and 8 of "Valbonne" from Oxford Univ.) shows that no correct inlier is obtained. Red points are correct inliers. Blue points are incorrect inliers, but are consistent with the estimated fundamental matrix. In our results, the outliers have been eliminated by the robust fundamental matrix estimation.

wide baseline frames is not restricted to translation and looming only, but also may be combined with some significant rotation. The part of 3D rotation of the camera (R_x or R_y) may be converted to stretch-shearing or translation after projection, but the rotation component around Z axis, R_z, remains as the in-plane rotation. In these complicated cases, this scheme cannot accurately recover point correspondences. Fig.5.d and e show that the correct matches cannot be obtained if there are some significant in-plane rotations.

4.3 Affine Matrix Decomposition

The reason for the above problems is that we minimize the error ϵ (Eq.3) using only the first order Tayor expansion and employing the Newton-Raphson iteration. This method is very sensitive to the initial state, which is usually set as $A =$ I, $d = 0$, $\mu = 1$, and $\delta = 0$. This initial state is only correct for a small in-plane rotation and scaling.

In order to find a reasonable initial state for the Newton-Raphson iteration, we decompose an affine matrix A into three components, rotation matrix R, scaling matrix S, and stretch-shearing matrix E using the Singular Value Decomposition (SVD).

$$\begin{aligned} A &= UDV' = U(V'V)DV = (UV')(VDV') \\ &= \begin{bmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{bmatrix} \begin{bmatrix} v_1 & v_h \\ v_h & v_2 \end{bmatrix} \\ &= \begin{bmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{bmatrix} \begin{bmatrix} \kappa & 0 \\ 0 & \kappa \end{bmatrix} \begin{bmatrix} \frac{1}{\kappa} & 0 \\ 0 & \frac{1}{\kappa} \end{bmatrix} \begin{bmatrix} v_1 & v_h \\ v_h & v_2 \end{bmatrix} \\ &= \begin{bmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{bmatrix} \begin{bmatrix} \kappa & 0 \\ 0 & \kappa \end{bmatrix} \begin{bmatrix} v_1/\kappa & v_h \\ v_h & v_2/\kappa \end{bmatrix} \\ &= R(\alpha)S(\kappa)E, \end{aligned} \quad (4)$$

where U and V are the orthogonal matrices, D is a diagonal matrix, and E is a positive symmetric matrix. In 2D spaces, U, V and $R(\alpha)$ are represented as rotation matrices, $S(\kappa)$ is a scaling matrix, E is represented as a stretch-shearing matrix.

After decomposing the affine matrix, it is clear that this affine matrix implies a non-linear rotation component, $R(\alpha)$, which depends on one component of the 3D camera rotation, R_z. The scaling matrix $S(\kappa)$ depends on the focal length (zooming) and 3D translation component T_z (looming) of the camera. Therefore, if the viewing angle is large or if there is some significant scaling between two images, it is very difficult to obtain the correct convergence by using the linear Tayor expansion with the Newton-Raphson iteration from the initial state, $A =$ I, $d = 0$, $\mu = 1$, and $\delta = 0$. Fig.6.c shows the image patches with a significant rotation or scaling that do not converge to the correct solution by using our corner matching scheme.

In order to obtain a reasonable initial state, we simplify the affine matching matrix $A \approx R(\alpha)S(\kappa)$. As a result, this affine matching space can be divided into independent spaces, rotation and scaling. Then, rotation angle α and scaling factor κ can be easily quantized into different ranges respectively.

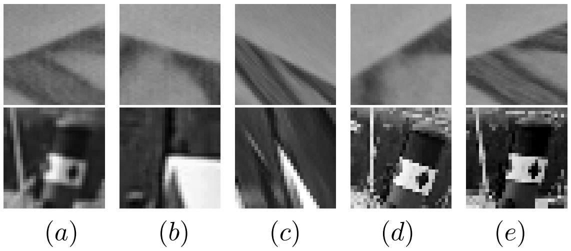


Figure 6: Matching procedures for two corresponding corners (30×30). Top: rotation case, the corners from “House” (Fig.5.c), bottom: scaling case, the corners from ‘Boat” (Fig.10.a). (a) and (b) are the windows around the original corners in two frames. (c) is the warped version of (b) using the corner matching scheme. (d) is the warped version of (b) after finding the best rotation and scaling in the first matching stage. (e) is the final result by using our two-stage matching algorithm. It is clear that our method maintains the appearance (compare (a) and (e)) of the patch over two wide baseline frames, and provides an accurate matching for the corresponding edge-corners.

4.4 Two-Stage Matching

Using this quantization property of the affine matrix, we design a two-stage algorithm for the corner matching on general wide baseline image pair.

The first stage is to find a reasonable initial state for the Newton-Raphson iteration. For any corner p_{1i} in I_1, we select p_{2j} from I_2 in a large search area. Then, we apply a rotation-scaling warping $R(\alpha)S(\kappa)$ to a window around p_{2j} and compare this window with a window around p_{1i} in I_1 by using SSD. We search for a minimal image residual by varying α from α_l to α_h. At each rotation angle, we use five different ranges of κ for the scaling matrix S, which are 4, 2, 1, 0.5 and 0.25. In our implementation, we use N-split tree (large step of 20° and small step of 4°) to speedup the search between α_l to α_h.

After computing the best estimates of matrices $R(\alpha)$ and $S(\kappa)$, we initialize the affine matrix $A = R(\alpha)S(\kappa)$ in Eq.2, which provides a reasonable gradient descent direction for the Newton-Raphson iteration. Then, following this gradient descent direction, the correct affine matching matrix A between two corners is determined and the residual is minimized.

The advantages of our approach are: First, we only compare corners with corners instead of searching every pixel in a big area, that may have several local minima. Second, the first step of our matching method quantizes affine space by using the rotation angle and scaling factor. It avoids the computation of a non-linear components $R(\alpha)$ and large scaling $S(\kappa)$ using gradients I_x and I_y, but recovers an approximate state of the geometric deformation due to the large rotation and scaling. Third, based on the reasonable initial value of $A = R(\alpha)S(\kappa)$, the second step follows the correct initial gradient descent direction and quickly finds an optimized solution of A between p_{1i} and p_{2j}, which efficiently avoids the local minima.

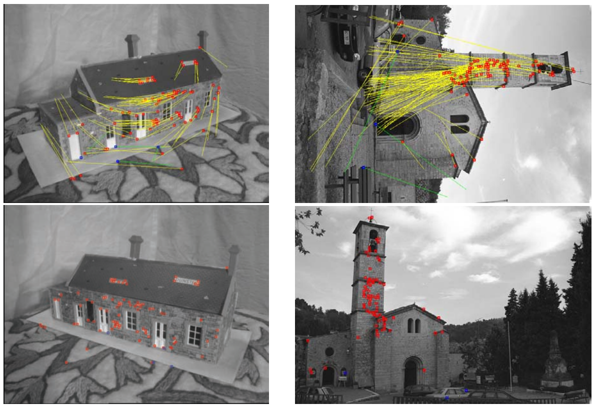


Figure 7: Results by our two-stage matching scheme. The left shows 83 correct inliers, several of them are on the ground. The right shows 78 correct inliers.

Fig.6 compares the results by using our two matching schemes, and illustrates the detailed step of our matching procedure. It is very clear to see that the correct initial state is very important to lead to a reliable optimized solution for correspondences matching. Fig.7 shows that using the two-stage matching algorithm, we can overcome the significant in-plane rotation problem and recover more reliable correspondences.

4.5 Increasing Correspondences by Using Epipolar Geometry

Even though we obtain many reliable correspondences over two images by our two-stage matching algorithm, some correspondences cannot be determined due to the self-similar structures and some false matches. These false matches are consistent with the estimated fundamental matrix and cannot be eliminated by using the robust fundamental matrix algorithm (blue points in Fig.5).

After robust fundamental matrix estimation, the epipolar geometry can be obtained. Let F denote the fundamental matrix, the epipolar line of p_{1i} on the second image is $l_{1i} = Fp_{1i}$. The corresponding point of p_{1i} should be located around this epipolar line on I_2. Therefore, for each point p_{2j}, we check the distance d_{1i2j}, such that

$$d_{1i2j} = \sqrt{(p_{2j}^T F p_{1i})^2 / \left((Fp_{1i})_1^2 + (Fp_{1i})_2^2\right)}. \qquad (5)$$

If the distance is less than threshold ξ ($\xi \in (2 \sim 4)$), we use two-stage matching algorithm to reestimate the residue ϵ_{1i2j} between these two points p_{1i} and p_{2j} in the two images. Since the probability of self-similar corners is dra-

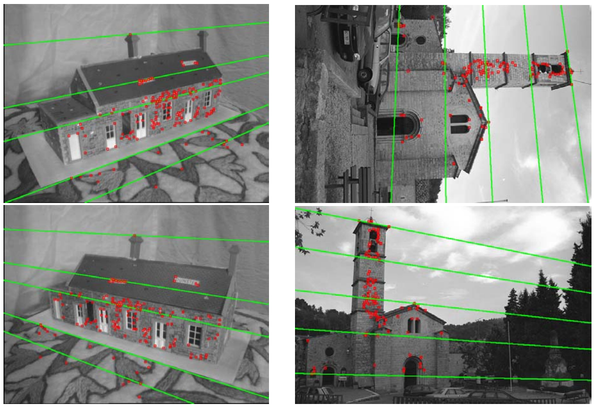


Figure 8: Refined results obtained by our approach. The left shows 150 correct inliers, and the RMS is 0.418. The right shows 102 correct inliers, the RMS is 0.632. Several green corresponding epipolar lines are drawn on the images.

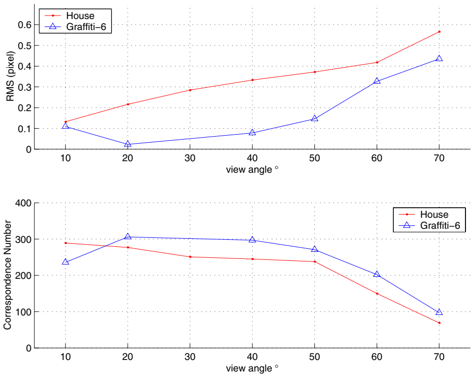


Figure 9: RMS and number of corresponding points of “House” and “Graffiti-6” sequences under different viewing angles.

matically reduced within a small band around the epipolar line, this method can efficiently increase the matches, which may be missed due to the self-similarity. Fig.8 shows the final matching results obtained by our approach.

5. Experimental Results

In this section, we first demonstrate our results on wide baseline matching. All of our experiments used gray images to compute correspondences, and most of the images are from the public domain resources (CMU, Oxford, OSU, INRIA, etc). Then, we illustrate several view morphing examples, where their epipolar geometries are estimated by proposed method, to show that our matching algorithm is effective and stable.

Fig.9 shows the results for two sequences of real scenes (Fig.1), where the viewing angle is changed from 0° to 70°. The RMS distance error increases as the angle increases, but is less than 0.6 pixel. After viewing angle reaches the 60°, the number of correct corresponding points decreases due to the severe occlusion and shrinking. Fig.10 shows the matching results for four different scenes, which include several kinds of significant camera motions (rotation, translation, or scaling) with some severe occlusion, illumination changes, and in presence of self-similarity structures. Fig.10.a shows four times scaling between the two images. Fig.10.b, c and d show that the non-inplane viewing angles are over 50°. Fig.11 shows the view synthesis results using our estimated matches and employing view morphing technique [15, 10].

Name (Frame No)	A-H	T-S (RMS)	RF (RMS)
Graffiti-5 (4,8)	33	276 (0.110)	299 (0.127)
Graffiti-6 (1,5)	27	135 (0.341)	202 (0.327)
Boat (0,5)	22	163 (0.301)	175 (0.560)
Valbonne (5,14)	14	57 (0.665)	185 (0.658)
Valbonne (9,13)	22	155 (0.486)	194 (0.661)
UBC (8,10)	34	184 (0.266)	227 (0.242)

Table 1: The comparison of Affine-Harris method [6] with our approach. A-H is the results using Affine-Harris method [6]. T-S is the results using two-stage matching without refinement. RF is the results after increasing correspondences using epipolar geometry.

We also tested a number of image pairs from INRIA and compared our results with the Affine-Harris method of [6] in Table 1. In all of the test cases, our method was able to determine more corresponding points than their method even without refinement step.

Note: All of these results are available on our web site: http://www.cs.ucf.edu/∼vision/projects/widematching/.

6. Conclusions

In this paper, we successfully solved the problem of how to obtain reliable corresponding points over two wide baseline frames. First, a number of affine-invariant edge-corners are detected in both images. Then, based on SVD decomposition of affine matrix, we found that the affine matching space between two corners can be approximately divided into two independent spaces by rotation angle and scaling factor. Using this crucial property of the affine matrix, we designed a novel two-stage matching algorithm to determine the robust matches between the edge-corners, which can effectively overcome the significant affine transformation in the wide baseline images. Moreover, employing the estimated epipolar geometry, we efficiently refined and increased the matches. Finally, we demonstrated that a series of virtual views can be correctly synthesized by using our correspondences and estimated epipolar geometries.

We have tested a number of wide baseline image pairs under different severe camera motions with illumination changes, occlusions, and self-similarities, and have obtained the excellent results for all of these cases.

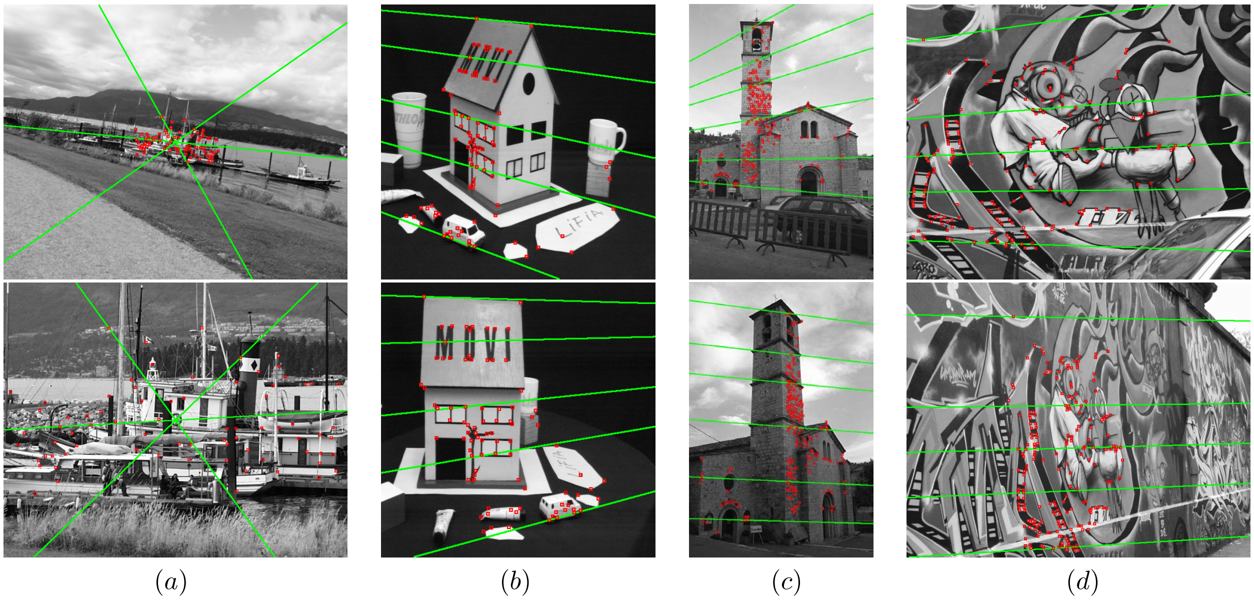


Figure 10: Final matching results by our algorithm. (a) (frame 8 and 0 of "Boat" from INRIA.) shows 75 inliers, the RMS is 0.686. (b) (frame 5 and 9 of "Movi" from OSU) shows 89 inliers, the RMS is 0.471. (c) (frame 5 and 14 of "Valbonne" from Oxford Univ.) show 185 inliers, the RMS is 0.658. (d) (frame 1 and 5 of "Graffiti-6" from INRIA) shows 202 inliers, the RMS is 0.327. Several green corresponding epipolar lines are drawn in each pair of images.

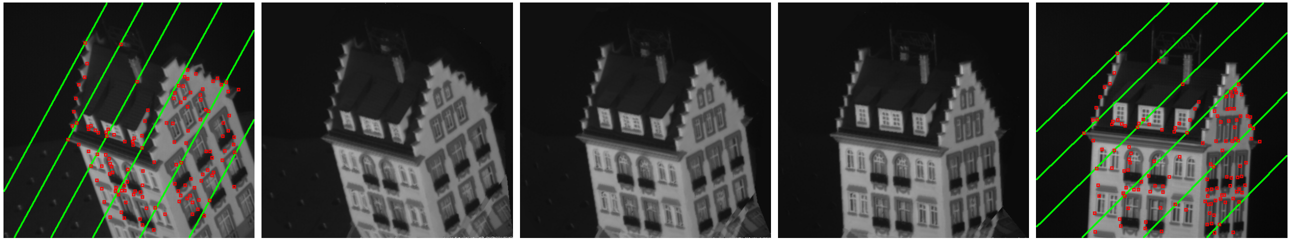


Figure 11: View synthesis results after matching for "Hotel"(frame 0 and 100 from CMU). There are 131 inliers, the RMS is 0.664.

References

[1] A. Baumberg. "Reliable feature matching across widely separated views". *CVPR*, 2000.

[2] O. Faugeras, Q. Luong. *The Geometry of Multiple Images*, The MIT Press , 2001.

[3] A. Fusiello, E. Trucco, T. Tommasini, V. Roberto. "Improving Features Tracking with Robust Statistics", *Pattern Analysis & Applications*, Vol. 2, pp. 312-320. 1999.

[4] R. Hartley, A.Zisserman. *Multiple View Geometry in Computer Vision*, Cambridge University Press, 2000.

[5] H. Jin, P. Favaro, S. Soatto. "Real-Time Feature Tracking and Outlier Rejection with Changes in Illumination", ICCV, 2001.

[6] K. Mikolajczyk, C. Schmid. "An Affine Invariant interest point detector", *Proc. ECCV*, 2002.

[7] P. Pritchett, A. Zisserman. "Wide Baseline Stereo Matching", *Proc. ICCV*, pp. 754-760, 1998.

[8] K. Rangarajan, M. Shah, and D. Brackle, "Optimal corner detector", *Computer Vision, Graphics and Image Processing*, Vol.48, 230–245, 1989.

[9] F. Schaffalitzky, A. Zisserman. "Multi-view Matching for Unordered Image Sets or How Do I Orangnize My Holiday Snaps?", *ECCV*, 2002.

[10] S. Seitz, C. Dyer. "View Morphing", *SIGGRAPH*, 1996.

[11] J. Shi, C. Tomasi, "Good Features to Track", *CVPR*, 1994.

[12] S. Smith, M. Brady. "SUSAN-a New Approach to low level image processing", *IJCV*, Vol. 23. pp. 45-78, 1997.

[13] F. Shen, H. Wang. "A Local edge detector used for finding corners", *Proc. ICICS*, 2001.

[14] T. Tuytelaars, L. Van Gool. "Wide baseline stereo matching based on local, affinely invariant regions". *BMVC*, 2000.

[15] J. Xiao, C. Rao, M. Shah,"View Interpolation for Dynamic Scenes", *Proc. EUROGRAPHICS'02*, 2002.

[16] Z. Zhang. "Determining the Epipolar Geometry and its Uncertainty: A Review", *IJCV*, Vol. 27, pp. 161-195, 1998.