Dense Correspondence Extraction in Difficult Uncalibrated Scenarios

Ruan Lakemond, Clinton Fookes, Sridha Sridharan
Image and Video Laboratory, Queensland University of Technology
GPO Box 2434, 2 George St, Brisbane, Queensland 4001
Email: {r.lakemond,c.fookes,s.sridharan}@qut.edu.au

***Abstract*—The relationship between multiple cameras viewing the same scene may be discovered automatically by finding corresponding points in the two views and then solving for the camera geometry. In camera networks with sparsely placed cameras, low resolution cameras or in scenes with few distinguishable features it may be difficult to find a sufficient number of reliable correspondences from which to compute geometry. This paper presents a method for extracting a larger number of correspondences from an initial set of putative correspondences without any knowledge of the scene or camera geometry. The method may be used to increase the number of correspondences and make geometry computations possible in cases where existing methods have produced insufficient correspondences.**

***Keywords*-local image features; uncalibrated method; epipolar geometry; dense matching;**

I. Introduction

Finding correspondences between two views of a scene is an essential part of solving many computer vision problems. When the two viewpoints are widely separated and the camera geometry is not known, wide baseline matching techniques are used to locate correspondences. These correspondences can then be used to compute the scene geometry, which in turn allows more detailed scene analysis. In difficult cases, even the state of the art wide baseline matching techniques cannot produce sufficient correspondences to compute a reliable estimate of the scene geometry. This paper presents a method for acquiring a dense set of correspondences in parts of the scene when wide baseline matching has provided only a few reliable correspondences and no knowledge of the camera geometry is available. The extra correspondences acquired using this method can make computing scene geometry possible.

The technique presented in this paper may be useful in networks of sparsely placed cameras, cameras that produce low resolution images and scenes containing few features in the commonly viewed region. These conditions are often present in surveillance camera networks, for example.

In wide baseline scenarios two views of the same scene may appear very different. Structures in each view may be subject to projective transformations, occlusion and may present with different intensity and colour due to differences in camera exposure and color balance settings. If the camera geometry is not known, finding correspondences can be a difficult task. The problem has been addressed by locating and extracting interesting local features from each view of a scene, computing a description of each feature and then matching the features between views based on their descriptions. A thorough review of feature extractors was recently published in [1]. The performance of various feature extractors and descriptors has been evaluated in [2], [3], [4].

In order to successfully match features across uncalibrated wide baseline views, the features must have the following properties:

- Features need to be distinct. A feature must be unique and contain sufficient information to distinguish it from other features.
- Features need to be extracted in a repeatable fashion despite the variations in appearance.
- The local image region around the feature must be normalised prior to computing a description in order to compensate for the appearance variations between views. Alternatively the description method must be invariant to a change in viewpoint.
- Feature descriptors must be robust enough to deal with slight variations in feature appearance, but distinct enough to allow discerning between different features.

One implication of the above requirements is that only a fraction of all the features in a scene are sufficiently unique and contain sufficient information to be matched correctly.

If a sufficient number reliable correspondences can be found between a pair of images, then it is possible to compute the epipolar geometry of the cameras using a robust estimation technique such as RANSAC and its derivatives [5], [6]. The epipolar geometry can be used to constrain the matching problem by matching a feature from one view only to the features in close proximity to the corresponding epipolar line in the other view. This reduces the search space and the ambiguity between features that are not unique, allowing more matches to be found.

It may not be possible to find a sufficient number of accurate correspondences to compute the epipolar geometry of a pair of views. In the rest of this paper, a method is developed to take advantage of a few putative correspondences to extract additional correspondences without any knowledge of the scene and camera geometry. It is then shown that this method is useful in recovering epipolar geometry in difficult scenarios.

978-0-7695-3866-2/09 $26.00 © 2009 IEEE
DOI 10.1109/DICTA.2009.19

II. Affine Feature Guided Dense Correspondence Extraction

A large amount of information is required for each feature to be successfully matched in a difficult scenario. It seems likely that each feature then contains in itself numerous smaller scale features. These smaller features do not provide sufficient information to be matched on their own, but could possibly still be exploited to establish correspondences. Given a pair of matched features, matching smaller features within the corresponding image regions should theoretically be simpler, since the search space is confined to the feature regions. The question of how additional correspondences can be extracted given a matched pair of features is addressed in this section.

A corresponding pair of affine covariant features provides not only a corresponding pair of point coordinates, but also an affine normalisation transform for each feature [7], [8]. The search for small scale correspondences can be greatly simplified by making use of the normalisation transforms, at least in a local area around each feature. Ideally, this normalisation maps the local image regions around the two features to a common coordinate frame. In practice there is usually some inconsistency between the normalisation transforms, since they were computed independently from the different images. Furthermore, images can only be perfectly aligned by a linear transform if the image is of a flat plane. In order to perform dense matching between the normalised image regions, the regions must first be aligned more accurately in order to remove any inconsistencies. An alignment algorithm for this purpose is treated in Section II-A.

The aligned patches do not correspond pixel for pixel since the images they were formed from my differ to a great extent and the structure surfaces they were imaged from may not be simply planar. Section II-B presents a method for reliably extracting small scale correspondences from aligned image patches that accounts for these problems.

In summary, the following procedure for extracting additional correspondences from a matched pair of features is proposed and discussed in this section:

1) Accurately align the pair of matched image regions.
2) Select candidate points in the aligned image regions to match.
3) Align selected points across the two image regions.
4) Project new correspondences back to original images.

A. Accurate Patch Alignment

Affine covariant feature extractors operate on the assumption that corresponding small local image regions in different views are approximately related by an affine transform [7]. This assumption will also be used here. Affine covariant feature extractors such as MSER [9], Hessian Affine and Harris Affine [8], [10] provide an affine normalisation transform for each feature. These normalisation transforms can be used as a good initial estimate for a transform that aligns two corresponding image regions to a common coordinate frame.

Each local feature provides the following transform in projective 2 space ($\mathcal{P}^2$):

$$\mathbf{H}_n = \begin{bmatrix} k\mathbf{R}\mathbf{A} & \mathbf{T} \\ \mathbf{0}^\top & 1 \end{bmatrix}, \tag{1}$$

where k is a scalar corresponding to the scale of the feature, $\mathbf{R}$ is a rotation matrix, $\mathbf{A}$ is an anisotropic scaling matrix with determinant equal to 1 and $\mathbf{T}$ is a translation vector that maps the feature centre to the origin. The transform $\mathbf{H}_n$ maps an ellipse circumscribing the feature to an unit circle entered at the origin.

Applying the normalisation transforms of a pair of corresponding features to their local image regions produces a pair of approximately aligned images. As mentioned previously, the alignment will not be perfect because the normalisation transforms were computed independently from very different image. Typical alignment errors will be discussed in terms of the transform components in Equation 1.

If the image region is symmetric to a significant degree then it is likely that the selected orientation of the two regions do not correspond. The robustness of the SIFT descriptor makes it possible to match features despite such an error. It is therefore possible that there is gross error in terms of region rotation.

The anisotropic scaling component of the affine transforms can be expected to be reasonably accurate. Larger regions may suffer from projective transformation that cannot be fully accounted for by anisotropic scaling, however this effect is usually negligible. Image regions that contain more than one plane cannot be aligned perfectly using only a linear transformation of $\mathcal{P}^2$ and hence will suffer from varying degrees of error on each plane.

The translation component of the affine transforms (or the feature location) can also be expected to be reasonably accurate. The most significant source of translation error is non planar regions of the image.

The image intensity can be significantly different due to different camera exposure settings and lighting conditions. Due to the local extent of the image region, a linear mapping of intensity is often sufficient to compensate.

The following alignment algorithm is proposed to align two corresponding regions with the above characteristics:

1) Prepare the template image patch.
2) Prepare the initial transform relating the target image to the template image.
3) Find an initial estimate for the rotation component of the transform.
4) Apply the inverse compositional image alignment algorithm to find a precise estimate of the transform.
5) Check that the results of alignment are reasonable.

Each step will be discussed in detail below and the process is illustrated in Figure 1.

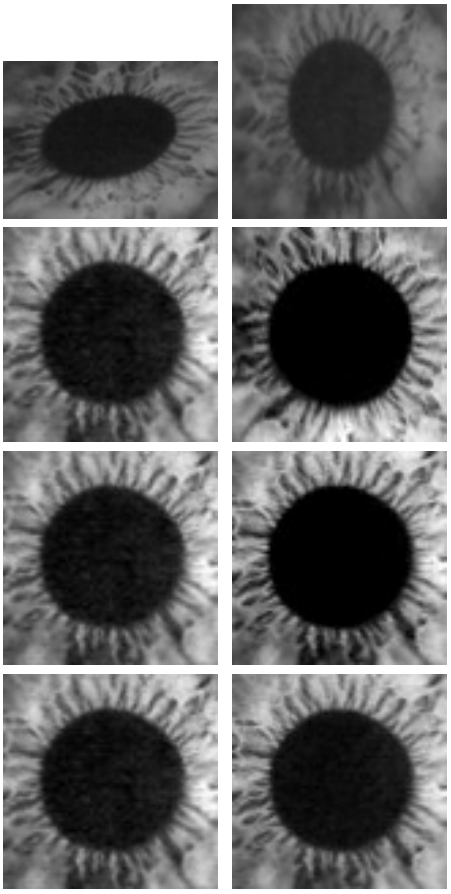

Figure 1. An example of the patch registration process. Line 1: original corresponding image regions; Line 2: patches after applying the feature normalisation transforms (steps 1 and 2); Line 3: patches after finding orientation (step 3); Line 4: patches after full affine alignment (step 4).

1) Template Preparation: Preparing the template image consists of selecting which of the regions to use as template and then normalising the template image. The choice of which feature to use as template is made based on the ratio of the eigenvalues q of the matrix $\mathbf{A}$ in the normalisation transform. The feature with q closest to 1 is chosen as template, since this feature's normalisation transform results in the least amount of deformation. The normalisation transform for the template region is then computed as,

$$\mathbf{H}_t = \begin{bmatrix} \alpha k_t \mathbf{A}_t & \mathbf{T}_t \\ \mathbf{0}^\top & 1 \end{bmatrix}, \tag{2}$$

with the constant α chosen such that the smallest eigenvalue of $\alpha k_t \mathbf{A}_t$ is 1. This prevents sub-sampling and aliasing of the source region. The rotation component $\mathbf{R}_t$ is ignored since it will be treated as unreliable in the following stages of the algorithm. A template image patch $P_t^{'}$ is then produced by transforming the original template image I_t according to $P_t^{'}(\mathbf{H}_t\mathbf{x}) = I_t(\mathbf{x})$. This places the normalised feature at the coordinate origin of the template patch. The template patch is only computed at a fixed number of samples in the coordinate range $\mathbf{x} \in \alpha k \left[[-1,1],[-1,1]\right]^\top$ (the local extent of the feature). An intensity transform for the template is computed from the minimum and maximum intensities of $P_t^{'}$ and used to generate the final template image patch P_t:

$$\begin{aligned} l &= \min\left(P_t^{'}\right), && (3)\\ h &= \max\left(P_t^{'}\right), && (4)\\ a_t &= \frac{255}{h-l}, && (5)\\ b_t &= -a_t l, && (6)\\ P_t &= f\left(a_t, b_t, P_t^{'}\right) = a_t P_t^{'} + b_t. && (7) \end{aligned}$$

2) Initialise Transform: The initial transform mapping the registration image to the template image patch is computed as,

$$\mathbf{H}_r = \begin{bmatrix} \alpha k_r \mathbf{A}_r & \mathbf{T}_r \\ \mathbf{0}^\top & 1 \end{bmatrix}, \tag{8}$$

where α is the same as in step 1 and $\mathbf{R}_r$ is once again ignored since it will be determined in the next step. The intensity transform is also initialised with the template patch's intensity transform parameters, $a_r = a_t$, $b_r = b_t$.

3) Initial Rotation Estimate: The rotation components of the feature normalisation transforms may have been estimated incorrectly by the feature extractor and should not be relied upon. An initial estimate for the rotation angle is obtained using the following method: The mean squared error between the template and the transformed target image is computed for rotation angles at regular intervals for a full rotation (20 intervals were used in the experimental implementation of this algorithm). A parabola is then fit to the minimum point in the error vector. The angle corresponding to the parabola apex is used as the initial rotation angle estimate.

4) Image Alignment: This step consists of applying the inverse compositional image alignment algorithm to simultaneously refine the parameters of the coordinate transform H_r and intensity transform parameters a_r and b_r. The details of the inverse compositional algorithm will not be repeated here in full, refer to [11] for a detailed discussion on the topic. Only the part of the algorithm pertaining to the intensity transform parameters will be explained below.

The goal is to align the intensities of the template image patch P_t and the corresponding target image patch $P_r(\mathbf{H}_t\mathbf{x}) = a_r I_r(\mathbf{x}) + b_r$ computed over the same range of coordinates as P_t. Assuming the spatial alignment is not grossly inaccurate, if a linear transform is applied to the intensity values of the template patch, the error between this patch and the registration patch is defined as follows,

$$e = \sum_{\forall \mathbf{x}} \left[a_r^{'} P_t(\mathbf{x}) + b_r^{'} - P_r(\mathbf{x})\right]^2, \tag{9}$$

where $a_r^{'}$ and $b_r^{'}$ are the inverse compositional parameter updates associated with a_r and b_r. To find the values of $a_r^{'}$ and $b_r^{'}$ that minimise this linear error function, set the partial derivatives to zero and solve for the parameters. The result is,

$$\begin{bmatrix} a_r^{'} \\ b_r^{'} \end{bmatrix} = \sum_{\forall \mathbf{x}} \begin{bmatrix} P_t^2(\mathbf{x}) & P_t(\mathbf{x}) \\ P_t(\mathbf{x}) & N \end{bmatrix}^{-1} \sum_{\forall \mathbf{x}} \begin{bmatrix} P_t(\mathbf{x}) P_r(\mathbf{x}) \\ P_r(\mathbf{x}) \end{bmatrix}, \quad (10)$$

where N is the number of pixels in the patch P_t. The matrix

$$\sum_{\forall \mathbf{x}} \begin{bmatrix} P_t^2(\mathbf{x}) & P_t(\mathbf{x}) \\ P_t(\mathbf{x}) & N \end{bmatrix}^{-1}$$

can be precomputed at the start of the alignment algorithm. At each iteration the intensity transform parameters are updated as follows,

$$a_{r(i+1)} = \frac{a_{r(i)}}{a_r^{'}} \quad (11)$$

$$b_{r(i+1)} = \frac{b_{r(i)} - b_r^{'}}{a_r^{'}} \quad (12)$$

5) Check Results: The inverse compositional alignment is not guaranteed to succeed, though it is likely to succeed in the majority of cases (even when aligning image regions that do not correspond to the same structure). In order to eliminate correspondence where alignment has failed, the final mean squared error is evaluated against a threshold and the transform's smallest eigenvalue is checked to be within reasonable bounds. Since the purpose of these checks are only to detect gross error, they need only be set at the extremes of reasonable values. A mean squared error threshold of 60 and a minimum eigenvalue of threshold of 0.005 were selected for the experimental implementation.

After registration, purely planar regions should be aligned nearly perfectly, with only projective deformation and image noise having a minor effect. Regions that are not entirely planar will suffer from parallax errors.

B. Dense Correspondence Extraction

It is desired to find matching points between pairs of aligned patches and to project these to the original images. It can not be assumed that each pixel in one patch corresponds to the same pixel in the other patch, however it can be assumed that the alignment error is on the order of a few pixels. In order to find exact point matches, small subregions around the points must be aligned.

The aperture problem plays a significant role when attempting to align small images. If a transform is to be computed that maps a region from one image to a corresponding region from another image, then the region must provide enough information to determine the transform unambiguously. Because only point correspondences are of interest and because the images are already reasonably well aligned, it is sufficient to only use a translation transformation to align points instead of an affine transformation. This simplifies the problem and reduces the amount of information required in the image, since a translation has only two degrees of freedom.

Finding points that can be aligned is easily done by finding corner structures in the image. A corner is defined as the junction of two lines or as a line with locally maximum curvature. Such a structure provides at least two edge structures (or regions of high gradient) that are not parallel. Each edge contributes one constraint towards accounting for the two degrees of freedom in a translation. Many methods exist for efficiently locating corner structures in an image. The experimental implementation locates corners by finding the local maxima of the determinant of Hessian operator at a single scale [12]. The objective is not to match corners as is done in wide baseline matching, but to select regions that can be accurately aligned. For this reason, corners are only extracted from one of the images in the pair of aligned image patches.

Ideally the smallest region possible would be used to align a selected point across images in order to limit interference from surrounding structures. Using a small number of pixels for image alignment can however result in poor error estimates (due to a small number of samples) and hence poor update estimates. The image alignment process can become unstable as a result. The amount of noise or distortion in the images will also affect the stability. Images containing more noise will require larger sample areas to be stably aligned. This limits the minimum size of image regions that can be aligned. The experimental implementation aligns patches as small as 9×9 pixels. This also coincides with the support region for the determinant of Hessian corner extractor.

The following procedure is proposed for finding and accurately aligning correspondences in aligned feature patches:

1) Extract an image region around each feature at the maximum common resolution.
2) Extract corner locations using the determinant of Hessian operator (at one scale only and in one image only).
3) Accurately align corner points by aligning progressively smaller sub-images.
4) Project detected corners back to the original images.

Each step will be discussed in detail below and the process is illustrated in Figure 2.

1) Extract Image Patches: First the normalised patches for each feature are extracted at the same resolution as the source image using normalisation transform $\mathbf{H}_1$ and $\mathbf{H}_2$. The larger of the two images patches is then filtered with a Gaussian filtering to prevent aliasing and then down-sampled so that it is the same resolution (and size) as the other image patch.

2) Extract Corners: A set of corners $\mathbf{c}$ is extracted by computing the determinant of Hessian function in one of the patches and finding the local maxima of the function. The scale parameter of the determinant of Hessian operator

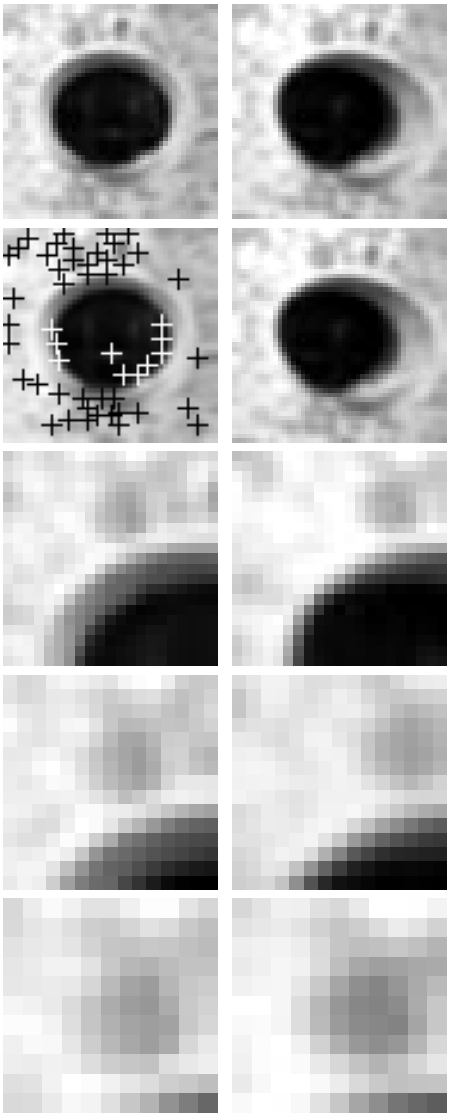

Figure 2. An example of the point registration process. Line 1: corresponding image patches after alignment (step 1); Line 2: corner points detected in image 1, indicated as black and white crosses (step 2); Line 3 to 5: progressively smaller aligned subregions around one of the points (step 3). Note the decreasing alignment error.

is set to 1 since only fine scale features are desired. Corners are not detected in both images since corner alignment will be performed using inverse compositional image alignment and not corner matching across images.

3) Align Corners: Small image patches around each corner are aligned by aligning progressively smaller patches, starting with a patch half the size of the normalised image patch. Only the translation component of the transform is modified during this step. This allows finer registration since it requires less image information (only two degrees of freedom) and is less prone to the aperture problem. It is considered sufficient to only modify the translation because previous alignment steps have already produced fairly well aligned image regions and because only the location of these fine scale points is required, not their full affine shape. This step usually requires few iterations of the registration algorithm. The template image in this step is arbitrarily chosen to be image 1. The resulting translation transform mapping the point in image 2 to image 1 is named $\mathbf{T}_r$.

4) Project Back: After alignment, each corner point is projected back to the original image coordinate frame.

$$\mathbf{c}_{1i} = \mathbf{H}_1^{-1}\mathbf{c}_i \quad (13)$$

$$\mathbf{c}_{2i} = \mathbf{H}_2^{-1}\mathbf{T}_r^{-1}\mathbf{c}_i \quad (14)$$

Figure 2 shows this process applied to an image region containing two different planes. Initially the alignment cannot be accurate for both planes and hence points on each plane are not correctly aligned. By aligning progressively smaller regions, points of interest on both planes may be aligned with great precision.

Figure 3 shows a comparison of attempting to compute the epipolar geometry from only correspondences generated using MSER and SIFT [13] and computing the geometry from a set of correspondences derived by applying the dense matching technique to the initial set of MSER correspondences. Because of the nearly planar scene geometry, there is a high likelihood of selecting a degenerate solution using only MSER features and RANSAC epipolar geometry computation. The result can be seen in Figure 3(*a-c*). The additional highly accurate correspondences provided by the dense matching technique results in the correct solution to the epipolar geometry with a large number of inliers (Figure 3(*d-e*)).

III. Algorithm Evaluation

Experiments were performed in order to establish whether the dense correspondence extraction algorithm presented in this paper can be of benefit in scenarios where the epipolar geometry cannot be recovered by conventional feature extraction and matching.

A. Experimental Setup

Test data sets were acquired using two cameras viewing a scene from different angles. Each set consists of images taken from a fixed camera configuration. See the following section for a description of the various camera arrangements and scenes used. Images captured at the maximum resolution provided by the cameras were used to compute the ground truth epipolar geometry and down-sampled versions of the images were used to perform tests. Each experiment applies various feature extractors in different combinations to a pair of images to find correspondences and to compute the epipolar geometry. The results of the various combinations of extractors is then compared. The focus will be on the difference in performance of extractors that make use of the dense correspondence extraction algorithm and those that do not.

(*a*) (*b*) (*c*) (*d*) (*e*)

Figure 3. Comparison of the results of computing epipolar geometry using MSER alone and MSER combined with dense matching technique. Images sourced from http://www.robots.ox.ac.uk/∼vgg/data1.html. Column (*a*) shows MSER features, (*b*) inlier correspondences after computing epipolar geometry from the set of MSER correspondences without dense matching, (*c*) the corresponding epipolar lines (note: degenerate), (*d*) inlier correspondences after computing epipolar geometry from the set of dense correspondences, (*e*) the corresponding epipolar lines.

(*1*) (*2*) (*3*) (*4*)

Figure 4. Example images from each dataset.

1) Test Data: Test data were acquired using pairs of digital cameras arranged to view a scene from widely separated views. Each set consisted of images taken from a pair of cameras in fixed position. The contents of the scene were altered for each pair of test images. Images were captured at maximum resolution (4.1 and 10 million pixel cameras were used) to allow ground truth computation. Test images were generated by scaling the images to 640×480 resolution. Scale factors were recorded for relating the test image geometry back to ground truth geometry.

The scenes were constructed such that they contained sufficient information for computing the epipolar geometry of the cameras, but in such a way that it might be difficult to extract this information automatically from low resolution images. Each scene contained features on 2 or more planes (scenes were not degenerate) visible from both cameras.

2) Feature Extraction and Matching: The following feature extractors were used:

- Harris Affine using the Harris & Stephens operator for point extraction [14], the Laplacian function and scale-space clustering for scale selection [15] and the second moment matrix for affine shape estimation [8] (labeled HaA).
- Hessian Affine using the determinate of the Hessian function for point extraction [12], the Laplacian function and scale-space clustering for scale selection [15] and the Hessian matrix for affine shape estimation [16] (labeled HeA).
- MSER [9].

Features were matched by computing a SIFT descriptor [13] for each feature and matching using one-to-one nearest neighbour matching with a threshold applied to the matching score. Matches with a second nearest neighbour within a threshold of the nearest neighbour were eliminated to prevent ambiguous matches.

3) Computing Ground Truth: Computing the ground truth epipolar geometry for a given dataset was performed using the following procedure:

1) Extract and match features across all high resolution image pairs using all feature extractors to be used in the experiments.
2) Compute an initial estimate of the epipolar geometry using RANSAC.
3) Match features again, this time using the initial geometry estimate to constrain matching.
4) Apply the dense correspondence extraction algorithm. Because the matches are at this stage verified by means of the epipolar constraint, dense matching will produce a large number of highly accurate correspondences and few outliers.
5) Compute a more accurate estimate of the epipolar geometry using RANSAC.
6) The set of inlier correspondences and the epipoles are scaled to match the resolution of the test images and are kept for use in error estimation during test trials.

4) Test Procedure: The procedure for each test trial consists of attempting to compute the epipolar geometry of a scene using a particular combination of feature extractors. The resulting geometry estimate is compared to the ground truth of the data set to determine the error in the estimate. A generous threshold is applied to the error to determine if a given estimate is sufficiently accurate to allow further geometry guided analysis of the scene. The computation may also fail as a result of insufficient correct correspondences. The number of trials that result in sufficiently accurate estimates are counted and compared across the different feature extractor combinations.

Two methods were used to measure the error in epipolar geometry estimates. The first method compares the epipoles of the estimate with those of the ground truth using a normalised distance:

$$e_e = \frac{|\mathbf{p}_{e1} - \hat{\mathbf{p}}_{e1}|}{|\mathbf{p}_{e1}|} + \frac{|\mathbf{p}_{e2} - \hat{\mathbf{p}}_{e2}|}{|\mathbf{p}_{e2}|}. \tag{15}$$

Here $\mathbf{p}_e$ indicates a ground truth epipole, $\hat{\mathbf{p}}_e$ indicates an epipole estimated during a test trial and $|\cdot|$ indicates the Euclidean norm such that $|\mathbf{x}| = (\mathbf{x}_1/\mathbf{x}_3)^2 + (\mathbf{x}_2/\mathbf{x}_3)^2$.

The second error metric is the Sampson distance [6] of the ground truth inlier correspondences evaluated using the estimated fundamental matrix:

$$e_s = \sum_i \frac{\left(\mathbf{x}_i^{'\top}\hat{\mathbf{F}}\mathbf{x}_i\right)^2}{\left(\hat{\mathbf{F}}\mathbf{x}_i\right)_1^2 + \left(\hat{\mathbf{F}}\mathbf{x}_i\right)_2^2 + \left(\hat{\mathbf{F}}^\top\mathbf{x}_i^{'}\right)_1^2 + \left(\hat{\mathbf{F}}^\top\mathbf{x}_i^{'}\right)_2^2}. \tag{16}$$

Here each $\{\mathbf{x}_i, \mathbf{x}_i^{'}\}$ is a corresponding pair of points from the set of ground truth correspondences, $\hat{\mathbf{F}}$ is the trial estimate of the fundamental matrix and $(\mathbf{v})_j^2$ is the square of the j-th entry of the vector $\mathbf{v}$.

A trial is considered successful if both error metrics are sufficiently low ($e_e < 1.0$ and $e_s < 100$). The number of successful trials for a particular combination of feature extractors gives an indication of how likely it is that the camera geometry can be recovered using these extractors on an unknown scene.

The following combinations of feature extractors were tested:

- Each extractor listed in Section III-A2 applied individually.
- All extractors combined.
- Each individual extractor combined with the dense correspondence extraction algorithm.
- All extractors combined with the dense correspondence extraction algorithm.

B. Results

Test results are presented in Table I. The table lists the number of trials for which the epipolar geometry was

Set details		Without dense matching				With dense matching			
Set No.	No. Frames	HaA	HeA	MSER	All	HaA	HeA	MSER	All
1	39	0	4	16	15	4	12	22	27
2	20	0	0	3	5	2	3	7	8
3	45	3	7	19	27	6	26	33	32
4	41	6	8	15	25	14	25	29	32
total %	190	4.7%	11.1%	30%	44.2%	15.3%	36.8%	56.3%	59.5%

Table I
TEST RESULTS: NUMBER OF SUCCESSFUL ATTEMPTS TO COMPUTE THE EPIPOLAR GEOMETRY USING VARIOUS COMBINATIONS OF FEATURE EXTRACTORS.

computed with sufficiently low error. Each column (except the first two) lists the results for an extractor combination. The first four extractor combinations do not make use of dense correspondence extraction and the last four do. It can be seen that making use of the dense correspondence extraction technique makes it possible to compute the epipolar geometry in many more cases as only using currently available feature extractors.

IV. CONCLUSION

The uncalibrated dense correspondence extraction method described in this paper is useful for extracting large numbers of small scale correspondences when only an initial set of correspondences is available. In difficult cases where it is not possible to automatically compute the epipolar geometry of a set of cameras due to insufficient accurate correspondences, using this technique can provide the additional correspondences required to succeed.

ACKNOWLEDGEMENTS

This project was supported by the Australian Government Department of the Prime Minister and Cabinet.

REFERENCES

[1] T. Tuytelaars and K. Mikolajczyk, "Local invariant feature detectors: A survey," *Foundations and Trends in Computer Graphics and Vision*, vol. 3, no. 3, pp. 177–280, 2008.

[2] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool, "A comparison of affine region detectors," *International Journal of Computer Vision*, vol. 65, no. 1-2, pp. 43–72, 2005.

[3] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence,*, vol. 27, no. 10, pp. 1615–1630, 2005.

[4] P. Moreels and P. Perona, "Evaluation of features detectors and descriptors based on 3d objects," *International Journal of Computer Vision*, vol. 73, no. 3, pp. 263–284, 2007.

[5] A. F. Martin and C. B. Robert, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[6] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. New York: Cambridge University Press, 2003.

[7] T. Lindeberg and J. Gårding, "Shape-adapted smoothing in estimation of 3-d depth cues from affine distortions of local 2-d brightness structure," in *Computer Vision – ECCV '94*, ser. Lecture Notes in Computer Science. Berlin / Heidelberg: Springer, 1994, vol. 800/1994, pp. 389–400.

[8] A. Baumberg, "Reliable feature matching across widely separated views," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, Hilton Head Island, South Carolina, USA, 2000, pp. 774–781.

[9] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide baseline stereo from maximally stable extremal regions," in *British Machine Vision Conference, BMVC 2002*, P. R. a. D.Marshall, Ed., vol. 1, Cardiff, UK, 2002, pp. 384–393.

[10] K. Mikolajczyk and C. Schmid, "Scale and affine invariant interest point detectors," *International Journal of Computer Vision*, vol. 60, pp. 63–86, 2004.

[11] S. Baker and I. Matthews, "Lucas-kanade 20 years on: A unifying framework," *International Journal of Computer Vision*, vol. 56, no. 3, pp. 221–255, 2004.

[12] P. Beaudet, "Rotationally invariant image operators," in *In International Joint Conference on Pattern Recognition*, Kyoto, Japan, 1978, pp. 579–583.

[13] D. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[14] C. Harris and M. Stephens, "A combined corner and edge detector," in *Alvey Vision Conference*, 1988, pp. 189–192.

[15] R. Lakemond, D. McKinnon, C. Fookes, and S. Sridharan, "A feature clustering algorithm for scale-space analysis of image structures," in *International Conference on Signal Processing and Communication Systems, ICSPCS'2007*, Gold Coast, Australia, 2007.

[16] R. Lakemond, C. Fookes, and S. Sridharan, "Affine adaptation of local image features using the hessian matrix," in *IEEE International Conference on Advanced Video and Signal Based Surveillance*, to appear, 2009.