3D Camera Pose Estimation using Line Correspondences and 1D Homographies

Irene Reisner-Kollmann[1], Andreas Reichinger[1], and Werner Purgathofer[2]

[1] VRVis Research Center
[2] Vienna University of Technology

Abstract. This paper describes a new method for matching line segments between two images in order to compute the relative camera pose. This approach improves the camera pose for images lacking stable point features but where straight line segments are available. The line matching algorithm is divided into two stages: At first, scale-invariant feature points along the lines are matched incorporating a one-dimensional homography. Then, corresponding line segments are selected based on the quality of the estimated homography and epipolar constraints. Based on two line segment correspondences the relative orientation between two images can be calculated.

1 Introduction

Matching features between two images is an important task in 3D computer vision, e.g. for camera parameter estimation, image retrieval or classification. Local, viewpoint invariant region descriptors are used for these tasks in many applications. They are independently extracted in input images and similar descriptors are selected as putative feature correspondences. Region descriptors are very robust for many scenes, but areas with distinctive textures are required [1]. In this paper we create correspondences between line segments instead of regions. This approach improves the results for scenes which contain few stable 2D features as caused by large homogeneous surfaces, but instead contain straight lines, e.g. urban scenes or interior rooms. The search space for corresponding features is decreased to salient line segments which in turn increases the distinctiveness of the feature descriptors.

Our algorithm matches the intensity profiles along line segments by matching distinctive feature points within these profiles. Corresponding feature points located along straight lines are correlated by a one-dimensional homography. This homography establishes an important constraint on the set of correspondending points. The similarity between two line segments is computed by comparing intensity values in the line profiles based on the estimated 1D homography. The final line segment correspondences are selected according to reprojection errors from a robust estimation of the camera parameters for the input images.

The advantage of our algorithm is that it doesn't rely on two-dimensional extremal features which may be sparse in some scenes. Splitting the problem

into a two-step approach reduces the dimensionality of the search space for correspondences. This has the advantage that there are fewer possibilities for corresponding points and the RANSAC estimator is more likely to find the best consistent matches.

The remainder of this papers is organized as follows. Section 2 describes the extraction of stable feature points along salient image lines. Section 3 presents how feature points are matched and how point correspondences are used for matching lines and estimating the relative orientation between two input images. Experimental results of our algorithm are presented in Section 4.

1.1 Related Work

Camera Pose Estimation A widely used approach for computing the relative pose of two cameras is the usage of locally invariant region descriptors [1–3]. Such region descriptors are invariant to a varying range of affine transformations in order to compare them in images from different viewpoints. Additionally, many descriptors are invariant to lighting changes. The detection of feature points in this paper is based on the localization of SIFT-features [2], but the generation of the scale space and the detection of extrema is reduced from three to two dimensions.

The relative camera pose is calculated in a similar approach as follows [4]: A set of feature points is extracted in both images and putative correspondences are detected by local descriptor matching. RANSAC [5] is used for a robust calculation of the camera parameters despite wrongly matched feature points. In each iteration of the RANSAC loop a minimum set of points needed for orienting the cameras is selected (5 points for calibrated cameras [6]). All other point correspondences are classified as inliers or outliers according to the estimated fundamental matrix. The camera parameters which returned the highest number of inliers are selected and all inliers are used for a final optimization of the parameters with a least squares solution.

Line Matching The goal of line matching algorithms is to find corresponding line segments between two or more images. Schmid and Zisserman [7] use known epipolar geometry for establishing point-to-point matches along line segments. Cross-correlation scores are computed for all point correspondences and their average is used as value for the line similarity. Cross correlation is adapted to large camera movements by approximating the areas next to line segments by planes and finding the best fitting homographies. This approach cannot be used for the camera orientation problem because camera parameters would have to be known in advance.

Bay et al. [8] combine color information and topological relations for matching line segments. An initial set of potential line matches is created by comparing the color histograms of stripes next to the line segments. As these line descriptors are not very distinctive, it is necessary to filter wrong matches based on topological relations. The topological filter takes three line segments or a combination of line segments and points and compares their sidedness in both images.

Meltzer and Soattao [9] match arbitrary edges across images instead of straight line segments with a similar approach to ours. They select key points at extremal values in the Laplacian of Gaussian along the edge. The feature descriptors for these points are based on gradient histograms similar to SIFT-points. The feature points are matched with a dynamic programming approach that uses the ordering constraint, i.e. corresponding feature points appear in the same order since projective transformations are order preserving. This constraint is used implicitly in our algorithm because a 1D homography maintains the order of points.

1D Point Correspondences Scale-space features in 1D signals have been used in other applications. Briggs et al.[10] match scale-space features in one-dimensional panoramic views. The images are taken by an omnidirectional camera used for robot navigation. A simple feature descriptor based on the value and curvature in the DoG-space is used for matching points which is sufficient for small camera baselines. The features are matched by circular dynamic programming, which exploits the fact that corresponding features in the circular images have to appear in the same order.

Xie and Beigi [11] use a similar approach for describing 1D sensor signals measured from human activities. The feature descriptors include the neighboring extremal positions of a key point. Corresponding points are found by nearest-neighbor matching.

2 Feature Extraction

In this section we describe how scale-invariant features along salient line segments are extracted from an image. The first step is to detect lines or line segments in an image, for which standard methods can be used. The next step is to create one-dimensional intensity profiles along the extracted lines. Finally, the scale spaces of these profiles are used to detect stable feature points.

2.1 Line Extraction

Although extracting lines from an image is not the main aspect of this paper, we want to depict some details about the line segments we use for matching. We use images that have been undistorted in a preprocessing step in order to contain straight lines. The parameters for undistortion can be computed together with the intrinsic camera calibration. For uncalibrated cameras, it is possible to use other undistortion methods, e.g. with a line-based approach [12].

Architectural scenes often contain collinear line segments, e.g. along horizontally or vertically aligned windows. Collinear line segments which correspond to the same 3D line in world space induce the same homography on a 1D line camera when they are matched to lines in another image. Therefore it is useful to extract lines across the whole image to get more robust homographies and point matches. On the other hand, parts of the line not corresponding to edge

features in the image should not contribute to the matching process, because it is possible that they are occluded by other objects. For this reason, we use one line segment for all collinear segments and apply weights according to the underlying image edge.

We use a Canny edge detector for producing an edge image and then apply a Hough transform for detecting straight lines. We sample the image along the lines with a fixed step size of one pixel and compute a weight for each sample point. The weights are based on the image gradients and denote the probability that a sample point is part of an image edge. The calculation of weights is defined in Equation 1 where g_i is the gradient at the sample point i and n is the normal of the line. The cosine of the angle between the gradient and the line normal is used to exclude sample points where another edge crosses the current line.

$$w_i = \|g_i\| \cdot \left| \frac{g_i}{\|g_i\|} \cdot n \right| = |g_i \cdot n| \tag{1}$$

For efficiency, low-weighted parts at the ends of a line segment are cut off and not used in subsequent operations. The rest of the line is kept for detecting and matching feature points. The weights are used for decreasing the impact of matched feature points at sample points with no underlying image edge.

2.2 1D Line Profiles

Straight lines can occur at edges within an object or at the border of an object. In the first case both sides of a line are quite stable with respect to each other. However, in the second case the images are only consistent on the side of the object, whereas the background changes with the viewpoint due to parallax. Therefore we investigate the two sides of the line separately during feature detection and matching.

For each side of a line, the image is sampled with the same step size as the weights in Section 2.1. In order to get a more descriptive representation we do not sample a single line but a rectangular extension of the line by a width w into the respective direction. This sampled rectangle is collapsed into a 1D profile by averaging the intensity values for more efficient subsequent computations.

An important parameter is the width w. It has to be large enough to contain distinctive information, i.e. it has to contain image parts next to the edge itself. Otherwise it is not possible to distinguish between noise and image features. If it is too large, on the other hand, multiple features may be collapsed and the one-dimensional profile is very smooth. Furthermore, the corresponding image parts next to a line segment may have differing widths in the case of large projective distortion. We used a profile width of 40 pixels in our experiments, but this parameter clearly depends on image resolution and scene contents.

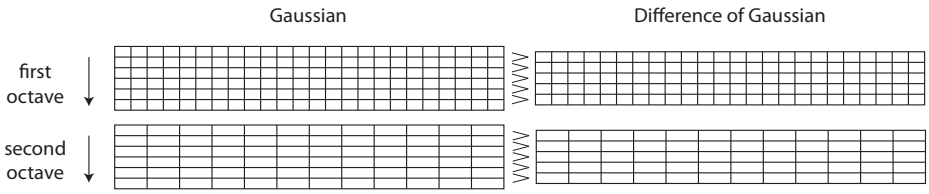


Fig. 1. Generation of Gaussian and DoG scale space for 3 scales per octave.

2.3 Scale Space Analysis

For each line profile a Gaussian scale-space is created in a similar manner as for two-dimensional SIFT-features [10, 2]. The one-dimensional signal $I(x)$ is convolved with a Gaussian filter $G(x, \sigma)$ over a range of different scales σ:

$$S(x, \sigma) = G(x, \sigma) * I(x), \text{ with } G(x, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-x^2/(2\sigma^2)} \tag{2}$$

Stable feature points are located at extrema in the difference-of-Gaussian (DoG) scale-space $D(x, \sigma)$. The difference-of-Gaussian is an approximation of the Lapacian of Gaussian and can be computed as the difference of two nearby Gaussian scales separated by a constant factor k:

$$D(x, \sigma) = S(x, k\sigma) - S(x, \sigma) \tag{3}$$

The creation of the scale space is implemented efficiently with multiple octaves where the profile is resized to half of its resolution for each subsequent octave. The scale factor σ of neighbored scales is separated by a constant factor $k = 2^{1/s}$ where s is the number of scales per octave. This means that the scale factor σ is doubled within one octave. In each octave $s + 3$ Gaussian scales are created in order to use s DoG scales for extremal value detection. The scale space creation is depicted in Figure 1 and an example can be seen in Figure 2.

Potential feature points are extracted at positions where the value in the DoG-space is higher or lower than its eight surrounding values. The exact position of an extremum is determined by fitting a quadratic function using the Taylor expansion of the scale-space $D(x, \sigma)$ [13]. The DoG-value at the extremum is used to reject unstable extrema with low contrast. Extrema with small curvature are also rejected as unstable. Similar to Briggs et al. [10] we calculate the curvature as the geometric mean of the second derivatives $c = \sigma\sqrt{|d_{xx}d_{\sigma\sigma}|}$. The multiplication with σ is necessary to get a scale-invariant curvature value.

3 Feature Matching

Feature matching is split into two parts: The first part matches feature points between two image lines. The second part searches for corresponding lines based on the feature point correspondences and epipolar constraints.

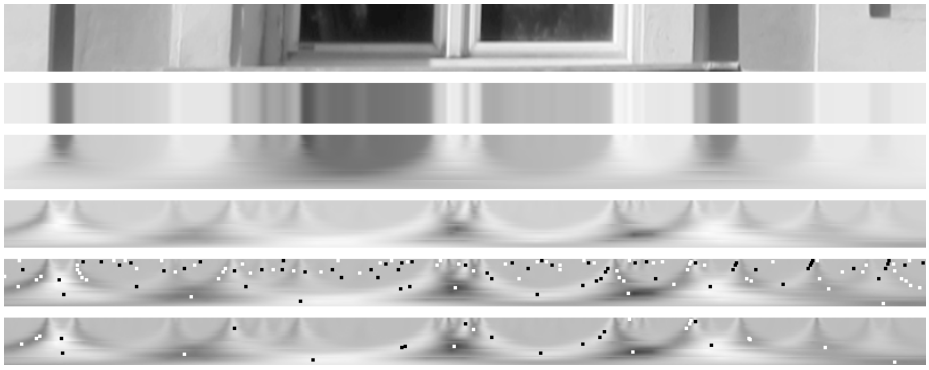


Fig. 2. From top to bottom: sampled rectangle next to a line segment, collapsed profile, Gaussian scale space with 6 octaves and 5 scales per octave, DoG scale space, extrema (white and black dots) in DoG scale space, accepted feature points with refined positions in DoG scale space.

As the orientation of the line segments is unknown, it is necessary to match each side of a line to both sides of the second line. In addition, the feature point descriptors themselves have to be invariant to the orientation of the underlying line. If a large number of line segments have to be matched or if time efficiency is important, it may be better to orientate all line segments such that the brighter profile is on the left side of the line [8]. The number of comparisons is reduced from four to two and the descriptor does not have to be changed. This approach fails if a line segment is located at the boundary of an object and the background contains large intensity changes.

3.1 Feature Point Descriptor

During feature point matching corresponding points between two line profiles should be detected. Therefore, we need a matching score that is a good estimate of the probability that two scale space features correspond to the same physical point in the world. Extrema of different types, i.e. minima and maxima, cannot correspond to the same object and a matching score of zero is assigned to these pairs.

Local properties used by Briggs et al.[10] were not discriminative enough in our experiments. Especially in the case of repetitive patterns, e.g. multiple windows of a building along a line, it was not possible to distinguish between correct and wrong correspondences.

We include the neighborhood of a feature point in order to increase the stability of the descriptor. Although the neighborhood can be quite different in case of large projective distortions or occlusions, it is a good description for many cases. The Gaussian scale space is sampled at neighboring points to the left and right of the feature point. The step size between sample points is based on the scale of the feature point in order to get a scale-invariant feature descriptor.

The matching score between two features is computed based on the sum of squared distances between corresponding neighboring samples. The matching scores allow to narrow down the set of potential feature matches, but it is still not discriminative enough to extract valid matches directly.

3.2 1D Homographies

Figure 3 shows the projection of a line in 3D space onto two images. The relation between the corresponding image lines $\mathbf{l_0}$ and $\mathbf{l_1}$ is not altered by rotations of the cameras around the 3D line $\mathbf{L}$. We rotate one camera such that the 3D line $\mathbf{L}$ and the image lines $\mathbf{l_0}$ and $\mathbf{l_1}$ are located in one plane. With this transformation the point matching problem is reduced to 1D cameras and it can be easily seen that corresponding points on the image lines are correlated by a one-dimensional homography. The 2×2-matrix $\mathbf{H}$ maps homogeneous points $\mathbf{x_i}$ on the first image line to the corresponding points $\mathbf{x'_i}$ on the second image line:

$$\mathbf{x'_i} = \mathbf{H}\mathbf{x_i} \tag{4}$$

This one-dimensional homography provides an important constraint on the set of corresponding points provided that two line segments belong to the same straight line in 3D space. The constraint that corresponding points appear in the same order is implicitly satisfied by the homography if the line segment is in front of both cameras. A minimum of three points is needed for the calculation of the homography, e.g. with the direct linear transformation (DLT) algorithm [14].

We use RANSAC [5] for the robust estimation of the homography. An initial set of potential point matches is generated by taking the N best correspondences for each feature point based on the feature descriptor presented in Section 3.1. In each iteration of the RANSAC loop three potential point matches are selected for the computation of a 1D homography. All other point correspondences are classified as inliers or outliers according to their symmetric transfer error e based on the absolute difference d (Equation 5).

$$e = d(\mathbf{x}, \mathbf{H^{-1}}\mathbf{x'})^2 + d(\mathbf{x'}, \mathbf{H}\mathbf{x})^2 \tag{5}$$

Finally, the resulting homography is optimized to all inliers and additional inliers from the other point matches are sought. Figure 4 shows an example for point correspondences between two line profiles and the associated homography.

A pair of corresponding lines has to fulfill two constraints in order to be accepted as line match. The first constraint is a minimum number of matched feature points. The second constraint tests how well the line profiles fit to the estimated 1D homography. Densely sampled points on the first line are transformed to their corresponding coordinate in the second line and vice versa. The sum of squared distances between the intensity values in the line profiles at corresponding sample points is used to measure the quality of the homography. The squared distances are multiplied by the weights obtained from the edge response in Section 2.1. The weighting is necessary to avoid contributions from image

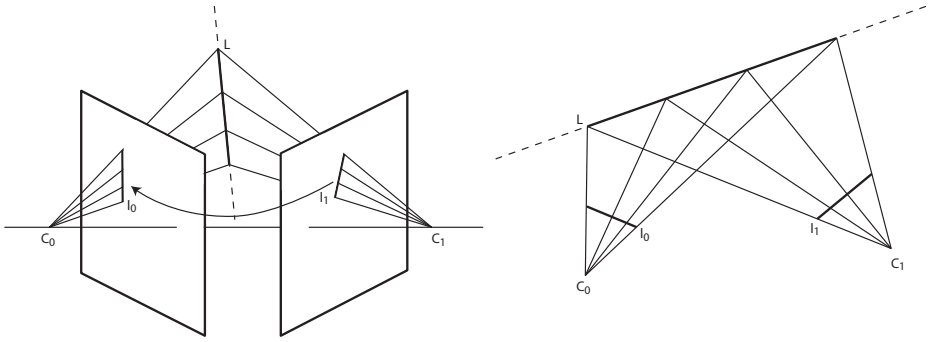


Fig. 3. Transformation from 2D-cameras to 1D-cameras. A 3D line segment L is projected onto two images with camera centers C_0 and C_1. The cameras are rotated such that L, l_0 and l_1 are located in a plane. In the right image can be seen that a 1D homography maps all points on line l_0 to their corresponding points on the second line l_1.

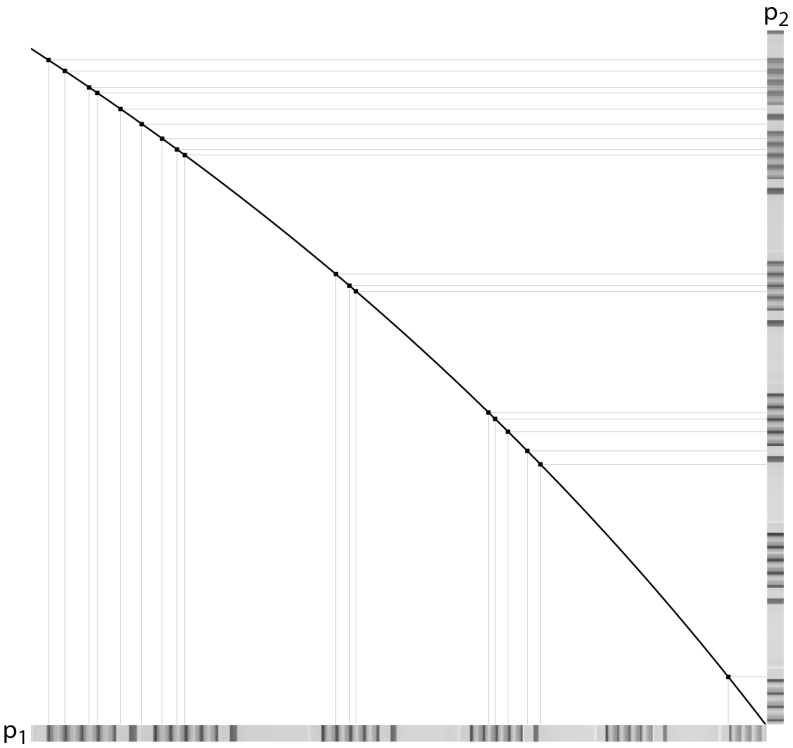
parts not belonging to the same 3D line, e.g. because the line is occluded by another object.

3.3 Camera Orientation

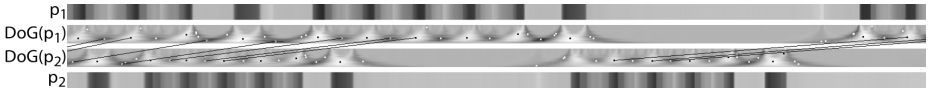
In the previous section a set of line matches together with point matches along these lines have been extracted. The point matches obey a one-dimensional homography, but the line matches do not fulfill any epipolar constraints yet. It is possible that there are wrongly matched line segments and that one line is matched to multiple lines in the other image.

Consistent line matches are extracted by computing the relative pose between the two images. In the calibrated case, two matched line segments together with their 1D homographies are required to compute the relative pose. We use the five-point-algorithm [6], for which three points are taken from the first line correspondence and two points from the second.

As there is usually only a rather small set of line matches, all potential line matches can be evaluated exhaustively. Of course a RANSAC approach could be used again to speed up computations if necessary. The relative camera orientation is computed for a pair of line matches. For all line matches 3D points are triangulated based on the estimated camera parameters for all point correspondences along the line segments. Line matches are classified as inliers respectively outliers depending on the reprojection errors of these 3D points. Additionally, it is evaluated if the 3D points are located on a three-dimensional line. In order to avoid degenerate cases, a line can only be classified as inlier once, although it might appear in multiple line matches. After evaluating all test cases, the relative orientation that led to most inliers is selected.



(a) Matched points and fitted 1D homography. Any point on one profile can be mapped to the other profile by the estimated homography.



(b) Part of the matched feature points shown in DoG-scale space.

Fig. 4. Feature point matches between two line profiles.

4 Experiments

We report results on four different example scenes. All images were taken with a *Canon EOS 5D Mark II* and have a resolution of 5616 × 3744 pixels. The scale spaces were created with four octaves and four scales per octave. The feature descriptors were created with 80 neighboring sample points.

The images of the first example show the facade of a house (Figure 5). Ten line segments were extracted in each images, from which correspondences between five line profiles were found initially. Four of these initial matches were the left and right profile of one corresponding line. The relative camera pose approved four line matches and 365 point matches.

Fig. 5. Facade scene: 365 point correspondences in four lines. Matching lines are displayed with the same color, unmatched lines are shown in white. The final point correspondences are visualized as black rings. Note that some lines are collapsed because they are located very near to each other.

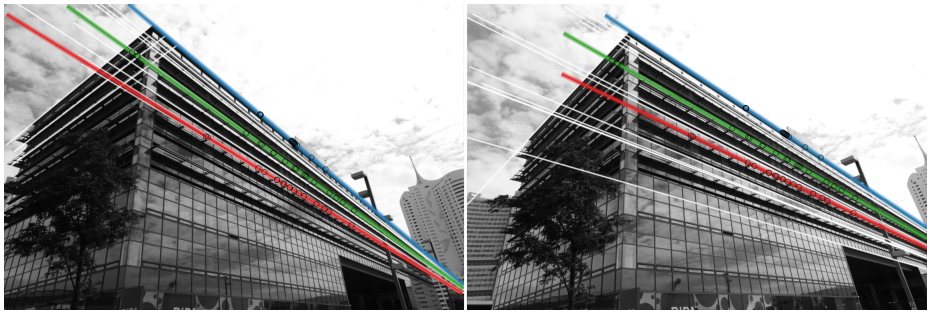


Fig. 6. Building scene: 52 point correspondences in two lines.

Figure 6 shows the pictures of another building. The initial line matching created three line correspondences which were all validated with 52 point correspondences by camera pose estimation. The problems in this scene are that there are many similar lines and that there are different occlusions from different viewpoints.

Figure 7 shows an interior scene. Ten line segments were extracted in both images. Two line correspondences and 41 point correspondences were validated by camera pose estimation from ten initial line profile matches.

The results for an urban scene can be seen in Figure 8. Ten line segments were extracted in both images, from which two line correspondence and 28 point correspondences were found and validated. This example shows that the algorithm is capable of matching lines at different scales.

5 Conclusion and Outlook

We have presented a new method for matching points located on line segments. The application of a one-dimensional homography allows to compute globally

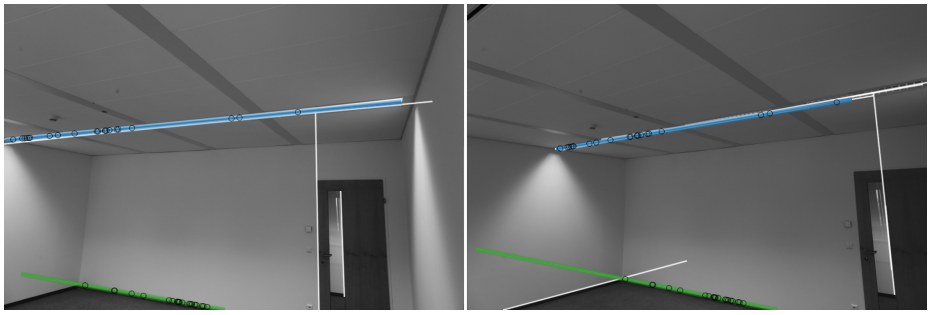


Fig. 7. Interior scene: 41 point correspondences in two lines.

Fig. 8. Urban scene: 28 point corrrespondences in two lines.

consistent point correspondences along the line segments. The set of corresponding points can be used together with a dense matching score for detecting corresponding line segments between the images. The set of potential line segments is evaluated based on the robust calculation of the relative pose.

We showed that the dimensionality of feature matching can be reduced by splitting it into point matching along line segments and line matching using epipolar constraints. The advantage of our algorithm is that feature points can be found although only a few distinctive 2D features are present in the images, provided that straight lines can be extracted.

For future work, we would like to use the estimated 1D homographies directly for calculating the relative pose between the input images. Two line correspondences and the associated 1D homographies could be used for estimating an initial solution to the camera pose problem. Another improvement of the algorithm will be a pre-selection of potential line matches before corresponding feature points are matched. This initial matching should be based on a simple line descriptor, e.g. a color histogram, and will increase the time efficiency of the algorithm.

References

1. Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. IEEE Transactions on Pattern Analysis and Machine Intelligence **27** (2005) 1615–1630
2. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision **60** (2004) 91–110
3. Matas, J., Chum, O., Urban, M., Pajdla, T.: Robust wide baseline stereo from maximally stable extremal regions. In: British Machine Vision Conference. (2002) 384–393
4. Snavely, N., Seitz, S.M., Szeliski, R.: Photo tourism: Exploring photo collections in 3d. In: SIGGRAPH Conference Proceedings. (2006) 835–846
5. Fischler, M.A., Bolles, R.C.: Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM **24** (1981) 381–395
6. Nistér, D.: An efficient solution to the five-point relative pose problem. IEEE Pattern Analysis and Machine Intelligence **26** (2004) 756–770
7. Schmid, C., Zisserman, A.: Automatic line matching across views. In: Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition. (1997) 666–672
8. Bay, H., Ferrari, V., Van Gool, L.: Wide-baseline stereo matching with line segments. In: Proceedings of the 2005 Conference on Computer Vision and Pattern Recognition. (2005) 329–336
9. Meltzer, J., Soatto, S.: Edge descriptors for robust wide-baseline correspondence. In: Proceedings of the 2008 Conference on Computer Vision and Pattern Recognition. (2008) 1–8
10. Briggs, A.J., Detweiler, C., Li, Y., Mullen, P.C., Scharstein, D.: Matching scale-space features in 1d panoramas. Computer Vision and Image Understanding **103** (2006) 184–195
11. Xie, J., Beigi, M.S.: A scale-invariant local descriptor for event recognition in 1d sensor signals. In: Proceedings of the 2009 IEEE International Conference on Multimedia and Expo, Piscataway, NJ, USA, IEEE Press (2009) 1226–1229
12. Thormählen, T., Broszio, H., Wassermann, I.: Robust line-based calibration of lens distortion from a single view. In: Proceedings of MIRAGE 2003. (2003) 105–112
13. Brown, M., Lowe, D.: Invariant features from interest point groups. In: British Machine Vision Conference. (2002) 656–665
14. Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision. 2nd edn. Cambridge University Press, ISBN: 0521540518 (2004)