

Novel Ways to Estimate Homography from Local Affine Transformations

Daniel Barath and Levente Hajder

MTA SZTAKI, Distributed Event Analysis Research Laboratory, Budapest, Hungary
{barath.daniel, hajder.levente}@sztaki.mta.hu

Keywords: Homography estimation, Affine transformation, Perspective-invariance, Stereo vision, Epipolar geometry, Planar reconstruction

Abstract: State-of-the-art 3D reconstruction methods usually apply point correspondences in order to compute the 3D geometry of objects represented by dense point clouds. However, objects with relatively large and flat surfaces can be most accurately reconstructed if the homographies between the corresponding patches are known. Here we show how the homography between patches on a stereo image pair can be estimated. We discuss that these proposed estimators are more accurate than the widely used point correspondence-based techniques because the latter ones only consider the last column (the translation) of the affine transformations, whereas the new algorithms use all the affine parameters. Moreover, we prove that affine-invariance is equivalent to perspective-invariance in the case of known epipolar geometry. Three homography estimators are proposed. The first one calculates the homography if at least two point correspondences and the related affine transformations are known. The second one computes the homography from only one point pair, if the epipolar geometry is estimated beforehand. These methods are solved by linearization of the original equations, and the refinements can be carried out by numerical optimization. Finally, a hybrid homography estimator is proposed that uses both point correspondences and photo-consistency between the patches. The presented methods have been quantitatively validated on synthesized tests. We also show that the proposed methods are applicable to real-world images as well, and they perform better than the state-of-the-art point correspondence-based techniques.

1 INTRODUCTION

Although computer vision has been an intensively researched area in computer science from many decades, several unsolved problems exist in the field. The main task of the research behind this paper is to discover the relationship among the affine transformation, the homography, the epipolar geometry, and the projection matrices using the fundamental formulation introduced in the pioneering work of Molnar et al. (Molnár and Chetverikov, 2014) in 2014. The aim of this study is to show how this theory can be applied to solve real-life computer vision tasks like estimating the homography and affine transformation between planar patches more accurately than it can be done by classical methods (Hartley and Zisserman, 2003).

A two-dimensional point in an image can be represented as a 3D vector, it is called the homogeneous representation of the point. It lies on the projective plane P^2 . The homography is an invertible mapping of points and lines on the projective plane P^2 . Other terms for the transformation include collineation, projectivity, and planar projective transformation. (Hart-

ley and Zisserman, 2003) provide a specific definition: A mapping $P^2 \rightarrow P^2$ is a projectivity if and only if there exists a non-singular 3×3 matrix H such that for any point in P^2 represented by vector x it is true that its mapped point equals Hx .

The correspondence can also be formalized for 2D lines as $l' H^{-T} l$ where line parameters can be written as vectors l and l' on the first and second images, respectively. If a point p lies on line l , the transformed location x' must lie on the corresponding line l' .

It is a very exciting fact that the concept of homography was already known in the middle of the last century (Semple and Kneebone, 1952).

There are many approaches in the field to estimate homography between two images as it is summarized in (Agarwal et al., 2005). At first, we have to mention the simplest method called Direct Linear Transform (Hartley and Zisserman, 2003) (DLT). In that case one wishes to estimate 8 unknown parameters of the homography H based on known point correspondences by solving an overdetermined system of equations generated from the linearization of the basic relationship $x' \sim Hx$, where the operator \sim means equality up to scale. The linearization itself

distorts the noise, therefore optimization for the original nonlinear projective equations gives more accurate results. This can be done by numerical optimization techniques such as the widely used Levenberg-Marquardt (Marquardt, 1963) optimization. However, the linear algorithms can also be enhanced if data normalization (Hartley and Zisserman, 2003) is applied first.

(Kanatani, 1998) proposed a method to minimize the estimation error within the Euclidean framework since the noise occurs in image coordinates and not in abstract higher dimensional algebraic spaces.

Obviously, there are many other ways to estimate the homography: line-based (Murino et al., 2002), conic-based (Kannala et al., 2006; Mudigonda et al., 2004), contour-based (Kumar et al., 2004) and patch-based (Kruger and Calway, 1998) methods exist. However, the matching of these features is not as easy as that of points. Nowadays, there are very efficient feature point matchers (Morel and Yu, 2009). Despite the fact that so many kind of homography estimation techniques available in the field, we have not find any dealing with local affine transformation-based homography estimation.

Application of homographies. There are many cases in computer vision where homography is required. First of all, one has to write about camera calibration (Zhang, 2000). If the homographies between the 3D chessboard coordinates and the projected ones is computed for several images, then the intrinsic camera parameters can be computed as it is proved by (Zhang, 2000).

Camera calibration is the process of determining the intrinsic and extrinsic parameters of the camera, where intrinsic parameters are camera-specific: focal length, lens distortion, and the principal point. Extrinsic parameters describe the camera orientation, and its location in 3D space.

Estimation of surface normals is also an important application of plane-plane homographies. If the homography is known between the images of a plane taken by two perspective cameras, then the homography can be decomposed into camera extrinsic parameters, the plane normal, and the distance of the plane w.r.t. the first camera (Faugeras and Lustman, 1988; Malis and Vargas, 2007). Molnar et al. (Molnár et al., 2014) and Barath et al. (Barath et al., 2015) showed that the affine transformation is enough in order to compute the surface normal, and it can be computed from the homography by derivating that as it is described in the appendix.

A very important application area of homography estimation is to build 3D models of scenes where relatively large flat planes are present. Typical ex-

ample for such tasks is the reconstruction of urban scenes that is a challenging and long-researched problem (Musialski et al., 2012; Tanács et al., 2014). Nowadays, 3D reconstruction pipelines uses point correspondences to compute the sparse (Agarwal et al., 2011; Pollefeys et al., 2008) or dense (Furukawa and Ponce, 2010; Vu et al., 2012) reconstruction of the scenes. However, patch-based approaches has recently proposed (Bódis-Szomorú et al., 2014; Tanács et al., 2014).

The main contributions of the paper are as follows. The first part of the paper deals with homography estimation when the fundamental matrix is unknown. In this case, the affine parameters can be calculated from corresponding patches in stereo images. *(i) It is described that the homography can robustly be estimated using the affine transformations.* In the second part, we focus on the presence of the known fundamental matrix. *(ii) It is proven that the homography can be calculated only from one point correspondence and the related affine transformation if the epipolar geometry is known.* Finally, a novel algorithm is described. *(iii) We show here that homography can be estimated using only two point correspondences and the neighboring image patches if the cameras are fully calibrated.*

2 METHODS TO ESTIMATE HOMOGRAPHY FROM AFFINE TRANSFORMATION

The main contribution of this paper is to introduce different techniques in order to estimate the homography if affine transformations are known at different locations. We also show here that more efficient estimators can be formed if the epipolar geometry is known as well. The main geometric terms and concepts are summarized in this section first.

2.1 THEORETICAL BACKGROUND

Homography and affine transformation. The standard definition of homography is applied here which was mentioned in the introduction: a homography H is the mapping $P^2 \rightarrow P^2$ which maps each vector $x_i^{(1)} = [u_i^{(1)} v_i^{(1)}]^T$ to its corresponding location $x_i^{(2)} = [u_i^{(2)} v_i^{(2)}]^T$ as $[u_i^{(2)} v_i^{(2)}]^T \sim H[u_i^{(1)} v_i^{(1)}]^T$. (The upper and lower indices denote the index of the current image, and the number of the current feature point, respectively.) (Molnár and Chetverikov, 2014) showed

that the affine transformation

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix} \quad (1)$$

can be expressed from the parameters of the homographies as it is discussed in the appendix. These four parameters are responsible for horizontal and vertical scales, shear and rotation. The last column of the affine transformation A gives the offset.

Extracting homography with fundamental matrix. Relationship, which is well-known from epipolar geometry (Hartley and Zisserman, 2003) led us to make estimation process easier, and decrease the DoF of the problem if the fundamental matrix is known. This relationship is formulated as follows (Hartley and Zisserman, 2003):

$$\left[e^{(2)} \right]_{\times} H = \lambda F \quad (2)$$

where $e^{(2)} = [e_x^{(2)}, e_y^{(2)}, 1]^T$ denotes the epipole in the second image, and λ is the scale of the fundamental matrix F . The operator $[v]_{\times}$ is the well-known matrix formula representing the cross product with vector v . Remark that the rank of matrix $[v]_{\times}$ is two, therefore the third row of the matrix can be determined as the linear combination of the first two ones.

The basic relationship defined in Eq. 2 shows how the knowledge of fundamental matrix decreases the DoF of the estimation problem. The last row is redundant as the rank of $[e^{(2)}]_{\times}$ is two. Therefore, only the first two rows contain useful information. They can be written as

$$\begin{bmatrix} 0 & -1 & e_y \\ 1 & 0 & -e_x \end{bmatrix} \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} = \lambda \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \end{bmatrix} \quad (3)$$

This equation shows that the DoF can be reduced to 3 since the elements in the first two rows of the homography can be expressed by those in the third one (h_{31}, h_{32} , and h_{33}), if the fundamental matrix is known:

$$\begin{aligned} h_{11} &= e_x h_{31} + \lambda f_{21} & h_{12} &= e_x h_{32} + \lambda f_{22} \\ h_{13} &= e_x h_{33} + \lambda f_{23} & h_{21} &= e_y h_{31} - \lambda f_{11} \\ h_{22} &= e_y h_{32} - \lambda f_{12} & h_{23} &= e_y h_{33} - \lambda f_{13} \end{aligned} \quad (4)$$

Remark that both the fundamental matrix and the homography are determined up to an arbitrary scale. Therefore, one scale is allowed to be set to an arbitrary value. In our algorithms, $\lambda = 1$.

If Equation 4 is substituted into the relationship of the DLT method ($p^{(2)} \sim Hp^{(1)}$), then the homography can be computed. Remark that one point pair

gives only one equation as the fundamental matrix reduces the DoF of the correspondence problem to one since the point pairs have to lie on the related epipolar lines. This homography estimation method is called 3PT in this study because the estimation can be carried out if at least three point correspondences (and the fundamental matrix) are given.

2.2 Homography estimation from Affine transformation (HA)

Based on the elements of the affine matrix a linear system of equations can be formed. The relationship between the affine transformation A_i belonging to the i -th point pair and the corresponding homography is discussed in the appendix. For the linearization, the Eqs. 9, 11- 13 have to be multiplied by the projective depth s (see Eq. 10). The obtained linear equations are as follows:

$$\begin{aligned} h_{11} - h_{31} \left(u_i^{(2)} + a_{i,11} u_i^{(1)} \right) & & (5) \\ -h_{32} a_{i,11} v_i^{(1)} - h_{33} a_{i,11} &= 0 \\ h_{12} - h_{32} \left(u_i^{(2)} + a_{i,12} v_i^{(1)} \right) & \\ -h_{31} a_{i,12} u_i^{(1)} - h_{33} a_{i,12} &= 0 \\ h_{21} - h_{31} \left(v_i^{(2)} + a_{i,21} u_i^{(1)} \right) & \\ -h_{32} a_{i,21} v_i^{(1)} - h_{33} a_{i,21} &= 0 \\ h_{22} - h_{32} \left(v_i^{(2)} + a_{i,22} v_i^{(1)} \right) & \\ -h_{31} a_{i,22} u_i^{(1)} - h_{33} a_{i,22} &= 0 \end{aligned}$$

Thus, the estimation can be written as a homogenous system of linear equations. However, all the elements of homography H cannot be estimated since elements h_{13} , and h_{23} are not present in the equations. This is trivial as these elements code the offset of the planes. Fortunately, the well-known (Hartley and Zisserman, 2003) Direct Linear Transformation (DLT) method can compute the offset as well. It gives two additional linear equations for the elements of the homography. They are as follows:

$$\begin{aligned} h_{11} u_i^{(1)} + h_{12} v_i^{(1)} + h_{13} - h_{31} u_i^{(1)} u_i^{(2)} - & & (6) \\ h_{32} v_i^{(1)} u_i^{(2)} - h_{33} u_i^{(2)} &= 0 \\ h_{21} u_i^{(1)} + h_{22} v_i^{(1)} + h_{23} - h_{31} u_i^{(1)} v_i^{(2)} - & \\ h_{32} v_i^{(1)} v_i^{(2)} - h_{33} v_i^{(2)} &= 0 \end{aligned}$$

Equations 5, and 6 give the linear relationship among the elements of the affine transformation, homography, and point locations. Six equations are obtained for each point correspondence. They can be written

as a homogeneous linear form $Bh = 0$ where the vector h , and matrix B contain the elements of the homography, and the corresponding coefficients, respectively. They are expressed in Eq. 7. The optimal solution (Björck, 1996) subject to $|h| = 1$ is obtained as the eigenvalue of $B^T B$ corresponding to the smallest eigenvalue. *If at least two point correspondences are given, the homography can be estimated.* This is a notable advantage of HA algorithm compared to the classical DLT one as the latter one requires at least four correspondences.

2.3 Homography estimation from Affine transformation with known Fundamental matrix (HAF)

In this section, we show that the estimation method becomes much simpler if the epipolar geometry is known. Equation 4 shows the basic relationship of the plane-plane homography and the epipolar geometry of the stereo camera setup. The affine transformation can be computed from the homography. (It is written in the appendix in detail.) By considering both relationship, the estimation of homography can also be written in a linear form even if the epipolar geometry is known. It is as follows:

$$\begin{aligned}
& h_{31} \left(a_{i,11} u_i^{(1)} + u_i^{(2)} - e_x \right) + \\
& h_{32} a_{i,11} v_i^{(1)} + h_{33} a_{i,11} = f_{21} \\
& h_{32} \left(a_{i,12} v_i^{(1)} + u_i^{(2)} - e_x \right) + \\
& h_{31} a_{i,12} u_i^{(1)} + h_{33} a_{i,12} = f_{22} \\
& h_{31} \left(a_{i,21} u_i^{(1)} + v_i^{(2)} - e_y \right) + \\
& h_{32} a_{i,21} v_i^{(1)} + h_{33} a_{i,21} = -f_{11} \\
& h_{32} \left(a_{i,22} v_i^{(1)} + v_i^{(2)} - e_y \right) + \\
& h_{31} a_{i,22} u_i^{(1)} + h_{33} a_{i,22} = -f_{12}
\end{aligned}$$

This is an inhomogeneous system of linear equations, thus it can be formed as $Cy = d$, where matrix C consists of the coefficients, $d = [f_{21}, f_{22}, -f_{11}, -f_{12}]$ while $y = [h_{31}, h_{32}, h_{33}]^T$ is the vector of the unknown parameters. The optimal solution in the least squares sense is given by $y = C^\dagger d$ where C^\dagger is the Moore-Penrose pseudo-inverse of matrix C . The elements of

matrix C are as follows:

$$\begin{aligned}
C_{11} &= \left(a_{i,11} u_i^{(1)} + u_i^{(2)} - e_x \right) & C_{12} &= a_{i,11} v_i^{(1)} \\
C_{13} &= a_{i,11} & C_{21} &= a_{i,12} u_i^{(1)} \\
C_{22} &= \left(a_{i,12} v_i^{(1)} + u_i^{(2)} - e_x \right) & C_{23} &= a_{i,12} \\
C_{31} &= \left(a_{i,21} u_i^{(1)} + v_i^{(2)} - e_y \right) & C_{32} &= a_{i,21} v_i^{(1)} \\
C_{33} &= a_{i,21} & C_{41} &= a_{i,22} u_i^{(1)} \\
C_{42} &= \left(a_{i,22} v_i^{(1)} + v_i^{(2)} - e_y \right) & C_{43} &= a_{i,22}
\end{aligned} \tag{8}$$

This method gives an overdetermined system for only one corresponding point pair, and an affine transformation. More equations can be added trivially to the system. It means, that *if one has only a single point pair and the related affine transformation, one is able to compute the homography.* Of course, it can be easily completed by other methods (e.g. DLT algorithm) exactly the same way as we showed in the previous section.

2.4 Improvements

Nonlinear refinement. The methods proposed here are solved by linear algorithms since the original problems are linearized by multiplying with the denominator. However, this multiplication distorts the original signal-noise ratio. If the denominator is relatively small, the distortion can be significant. For this reason, the nonlinear version of the proposed algorithms have to be formed. We used the classical Levenberg-Marquardt (Marquardt, 1963) numerical technique in order to compose nonlinear methods. To distinguish the linear and nonlinear versions of the methods, the names of the linear versions begin with 'LIN'.

Normalization. Normalization of the input data is usual in homography estimation (Hartley and Zisserman, 2003). Here we show how the normalized coordinates and the normalized affine transformation can be obtained.

Let us denote the normalizing transformations which are applied to the 2D point clouds in each image with T_1 , and T_2 . The normalized points are calculated on the first, and second images as $p_i^{(1)'} = T_1 p_i^{(1)}$, and $p_i^{(2)'} = T_2 p_i^{(2)}$, respectively.

It is not enough to normalize only the points, both the fundamental matrix and the affine transformations have to be normalized. The normalization formula for the fundamental matrix can be written (Hartley and Zisserman, 2003) as $F' = T_2^{-T} F T_1^{-1}$.

The affine transformations can also be normalized, as it is described in the appendix in details.

To distinguish the normalized versions of the

$$\begin{aligned}
B = & \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & -\left(u_i^{(2)} + a_{i,11}u_i^{(1)}\right) & a_{i,11}v_i^{(1)} & -a_{i,11} \\ 0 & 1 & 0 & 0 & 0 & 0 & -a_{i,12}u_i^{(1)} & -\left(u_i^{(2)} + a_{i,12}v_i^{(1)}\right) & -a_{i,12} \\ 0 & 0 & 0 & 1 & 0 & 0 & -\left(v_i^{(2)} + a_{i,21}u_i^{(1)}\right) & -a_{i,21}v_i^{(1)} & -a_{i,21} \\ 0 & 0 & 0 & 0 & 1 & 0 & -a_{i,11}u_i^{(1)} & -\left(v_i^{(2)} + a_{i,22}v_i^{(1)}\right) & -a_{i,22} \\ u_i^{(1)} & v_i^{(1)} & 1 & 0 & 0 & 0 & -u_i^{(1)}u_i^{(2)} & -v_i^{(1)}u_i^{(2)} & -u_i^{(2)} \\ 0 & 0 & 0 & u_i^{(1)} & v_i^{(1)} & 1 & -u_i^{(1)}v_i^{(2)} & -v_i^{(1)}v_i^{(2)} & -v_i^{(2)} \end{bmatrix} \\
h = & [h_{11}, h_{12}, h_{13}, h_{21}, h_{22}, h_{23}, h_{31}, h_{32}, h_{33}]^T
\end{aligned} \tag{7}$$

methods, the names of the normalized algorithms begin with 'Norm.'

Robustification. It is unavoidable in real application that the input dataset contains both inliers and outliers. We apply the RANSAC (Fischler and Bolles, 1981) paradigm in order to make the proposed methods robust. The names of RANSAC-based methods contain the word 'RSC'.

2.5 Theoretical contribution

It can be seen from the theory of HAF algorithm that if the fundamental matrix is known, then the homography and the affine transformation can unequivocally be calculated from each other in an observed point. *This property of perspective projection states that affine-invariance is equivalent to perspective-invariance if the epipolar geometry is known between stereo images.* In order to take the advantages of this property, fully calibrated camera setup is not needed, only the fundamental matrix between the cameras is required.

3 Homography Estimation based on Photo-consistency and Point Correspondences (RHE – Rotary Homography estimation)

The homography estimation (Agarwal et al., 2005) can be carried out using usual features in images such as point or line correspondences. Another approach is to use pixel intensities to estimate the plane-plane transformation between image patches (Habbecke and Kobbelt, 2006; Z. Megyesi and D.Chetverikov, 2006; Tanács et al., 2014). The study of (Habbecke and Kobbelt, 2006) proposes to estimate the four spatial plane parameters, while (Z. Megyesi and D.Chetverikov, 2006) and (Tanács et al., 2014) reduce the DoF of plane es-

timization problem to three using rectified images. Remark that rectification can be carried out if the fundamental matrix is known, the two projection matrices themselves does not have to be known.

Other possible solution is to use point correspondences in order to compute the homography (Hartley and Zisserman, 2003). If the fundamental matrix is known, the estimation can be calculated from three correspondences. If the epipolar geometry is not known, four points are required at least.

We show here that homography can also be estimated if both point correspondences and photo-consistency are considered. For the algorithm proposed in this section, two point correspondences are taken. The projection matrices of the stereo images are known, therefore, the spatial coordinates of the two points can be calculated via triangulation (Hartley and Sturm, 1997).

It is trivial that three spatial points determine the plane as they are enough to determine the homography. Two of those are calculated by triangulation, the remaining task is to determine the third one. The DoF of the problem is only one since an angle α ($\in (0, \pi)$) determines the plane as it is visualized in Fig. 1. This angle is determined via a brute-force (exhaustive) search in our approach. For each candidate value α , a spatial patch can be formed that consists of the two triangulated points p_1 and p_2 , and the angle of the patch is α . The cameras are calibrated, therefore the homographies between the projected patches can be calculated. Then this homography is evaluated. Its score is calculated as the similarity¹ of the corresponding pixels around the projected locations of points p_1 and p_2 . (The pixel correspondences are obtained by the homography.) The 3D patch with the highest similarity score gives the best estimation for the 3D patches. The obtained homography is determined by this 3D patch.

The proposed algorithm is as follows:

¹We use normalized cross correlation (Sonka et al., 2007) (NCC) for this purpose.

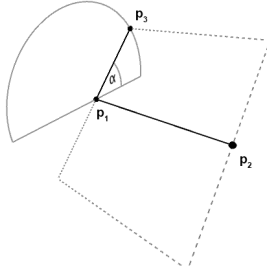


Figure 1: Rotating plane.

1. Calculate point p_3 related to the current α value, and the homography H_α . Then for the i -th ($i \in [1, 2]$) point pair compute $A_{\alpha,i}$ between the vicinities of the point projection using H_α .
2. Compute the similarity (NCC) related to each point and affine transformation. If the sum of the similarities in the two observed points is greater than the currently best candidate then $\alpha_{opt} := \alpha$.
3. If $\alpha < \pi$ increase α , and continue from Step 2. Otherwise, terminate with H_α .

4 EXPERIMENTAL RESULTS

The proposed homography estimators are tested both on synthesized data and real world images.

4.1 Test on Synthesized Data

The main goal of the tests is to generate different cases where homographies have to be estimated. For this reason, a stereo image pair represented by projection matrices is generated first. Their orientation are constrained, and the positions are randomized² on a 30×30 plane that is 60 unit far from the origin on axis Z . The generated cameras look at the origin. The remaining one DoF of the camera orientation is randomized as well. Then a 3D plane is generated at the origin with a random normal vector. Then 50 points are randomly sampled, and they are perspectively projected onto the two cameras. The ground truth homography between the projections of the plane is calculated as well.

The error values are defined as the average/median reprojection errors of the points.

All the proposed methods are tested³ in the syn-

²We applied zero-mean Gaussian noise for random number generation.

³All the tests have been implemented both on Matlab, and C++. It can be downloaded from webpage <http://web.eee.sztaki.hu>

thesized environment except for the RHE which requires real images for photo-consistency calculation. For each test, 100 different planes are generated on every noise level.

The proposed methods are compared to the OpenCV-implemented ‘findHomography’ function, which is a normalized DLT algorithm (Hartley and Zisserman, 2003) and a refinement stage using Levenberg-Marquardt algorithm (Marquardt, 1963) that minimizes the reprojection error. The other rival method is the normalized 3PT. The latter one is implemented by us.

Test with noisy point coordinates. In the first test case, 2D point coordinates are contaminated by zero-mean Gaussian noise, but the affine transformations do not. Two kinds of methods can be seen in the left plot of Fig 2: the first one uses the fundamental matrix, the second one does not. Within the first group, it can be observed that normalized HA performs better than OpenCV implementation. The second group which uses the fundamental matrix consists of HAF algorithm and the normalized three point (3PT) method. It can be seen that HAF performs significantly better.

Test with noisy affine transformations. The next test case (left plot of Fig. 2) is with noisy affine transformations. Noise in the affine transformation appears as a nearly identity random transformation. Every affine matrix is multiplied with such a transformation. Note that the horizontal axis in the charts shows only the noise of point coordinates. It can be seen that the original HAF is very sensitive to the affine noise, however, its RANSAC version balances this behaviour.

In the top plot of Fig 3, the variants of HA can be seen with contaminated point coordinates. It is evident that the normalized, numerically refined version gives the most accurate result. The bottom figure shows the different versions of the HAF. The normalized HA is also visualized for the sake of comparison. The average error seems to be very chaotic, however, the numerically refined version seems to be the best.

It is unequivocal that the proposed methods give more accurate results than the rival ones. Without the knowledge of the epipolar geometry, the normalized version of HA method performs better than the numerically refined normalized DLT. All methods is outperformed by HAF.

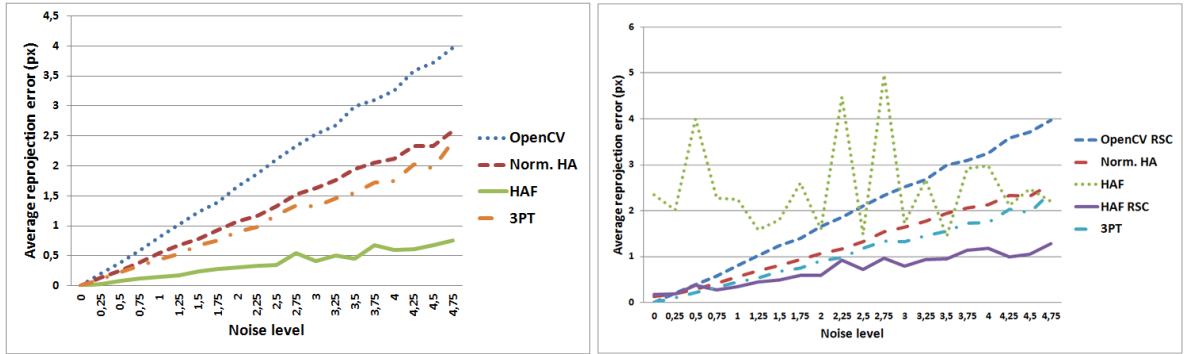


Figure 2: The left and right plots shows the average errors of the methods with noisy point coordinates and noisy affine transformation, respectively. The vertical axis are the average reprojections errors in pixels. The horizontal ones are the σ (spread) of the Gaussian-noise added to point coordinates. Affine error appears by multiplying the original affine transformation with a relatively small random transformation.

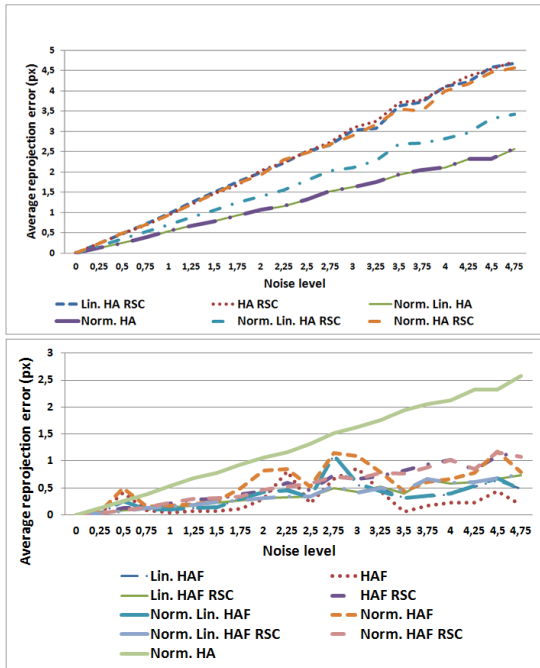


Figure 3: The average reprojection errors of the variants of HA, and HAF method are shown in the top, and bottom rows, respectively. The point are contaminated by Gaussian-noise, which σ value is denoted by the horizontal axis. The vertical one shows the average error in pixels.

4.2 Test on Real Data

Our algorithms are tested on the sequences of Oxford dataset ⁴.

Calculation of the affine transformation for real tests. In order to apply the proposed algorithms to real

⁴Dataset can be downloaded from <http://www.robots.ox.ac.uk/~vgg/data/data-mview.html>

data, the knowledge of the affine transformation is required between every single point correspondences. There are several ways to compute the affine transformation: brute-force algorithms, or affine invariant feature trackers (Mikolajczyk and Schmid, 2004). During our experiments the following method is used:

- (1.) Big planar surfaces are segmented using sequential RANSAC. For each planar patch the contained 2D point cloud are triangulated by Delaunay triangulation (B., 1934; Lee and Schachter, 1980).
- (2.) Then, for each point pair we iterate through all the corresponding triangles. The homography is computed between every triangle pair (on the first and the second images) using 3PT method, then the affine transformation is decomposed from that as it is described in the appendix.
- (3.) This method computes many slightly different affine transformations for every single point pair. Remark that all of them are used during the homography estimation as an overdetermined system of equations.

To visualize the quality of the proposed algorithms, the surface normals are computed, and they are drawn into the images. There are several normal estimators (Faugeras and Papadopoulos, 1998; Malis and Vargas, 2007; Barath et al., 2015) in the field, we choose the method of Barath et al. (Barath et al., 2015) due to its efficiency and simplicity. This estimator calculates the surface normal from the affine transformation related to the observed point instead of the homography in order to avoid the ambiguity of the homography decomposition (He, 2012).

The Oxford data set contain point correspondences, but we use ASIFT method (Morel and Yu, 2009) to detect and track points instead of the original data. The Hartley-Sturm triangulation (Hartley and Sturm, 1997) is applied to each point pair first. Planar regions are selected using sequential RANSAC,

however, it could be done by J-Linkage (Toldo and Fusiello, 2010), or other multi-homography fitting algorithm as well. Then fundamental matrix are calculated by the RANSAC 8-point technique (Hartley and Zisserman, 2003). The tests are both qualitatively and quantitatively evaluated. For the latter one, the error values are calculated as follows: 50% of the point correspondences are separated, and the homography is computed using only them. Then the reprojection error of the homography is computed for all the features. The final value was the RMS (Root Mean Squares) of the errors.

Another error metric have to be used for testing the RHE method. RHE computes the homography from only two feature correspondences. Therefore, the edges of the mentioned Delaunay triangulation are chosen as point pairs, and the homography related to the pair is computed by RHE. Then the reprojection error of every homography is calculated for all the feature points. The final reprojection error of the method is calculated as the average of these errors. In the following comparisons, the minimum reprojection error is also shown. Note that photo-consistency calculation processed on patches are of sizes from 60×60 up to 120×120 .

Figure 4 shows an example that demonstrates how the homography can be estimated by the proposed methods using many feature points. In this example, the baseline of the stereo setup is short, and the two main walls are segmented. The list of the obtained reprojection errors is written in Table 1. It is clear that the proposed algorithms (HA and HAF) outperform the rival ones (robust version of 3PT and OpenCV methods). HAF gives more accurate reconstruction than HA since it uses the fundamental matrix as additional information for the estimation. The obtained surface normals are perpendicular to each other (see the bottom of Fig. 4) as it is expected.

Delaunay triangulation is applied to the points of each wall (see the top of Fig. 4). Then RHE algorithm runs on every edge. The reprojection error of each estimated homography is calculated w.r.t. to every point pair selected from the current planar patch (both for the walls 'Left' and 'Right'). From the average of these reprojection errors, this algorithm yields less accurate results since it is calculated using only two point pairs. Even so, we have many estimated homographies (one for each edge of the triangulation) and we choose the one with the lowest reprojection error. It is turned out that it provides accurate estimation. Its results are the best and second one among all the other methods for the 'Left' and 'Right' walls, respectively.

The next two examples are seen in Figures 5 and 6.



Figure 4: The top row visualize the delaunay triangulation of the points. The bottom row shows reconstructed surface normals using homography of large walls on sequence 'College'.

Table 1: Reprojection errors (px) for sequence 'College'

	Left	Right
OpenCV RSC	3.824	2.668
3PT RSC	3.586	2.604
HA RSC	3.589	1.759
HAF RSC	3.585	1.677
RHE AVG	7.881	8.768
RHE MIN	3.442	1.692

The first one is the sequence 'Model House'. The segmentation finds two large planes on the scene: the wall and the ground. The next normal reconstruction example is called the sequence 'Library'. Two large planes are found in this scene: the wall and the roof. Then the proposed and rival homography estimators are applied. The normals reconstructed by the RHE algorithm are visualized in these figures, therefore the estimated normals are independent of each other.

The proposed and rival homography estimators are compared in Table 2. (Note that the patch size of the RHE algorithm was set to 60×60 for sequences 'Building' and 'Model House'.) It is clear that the proposed methods outperform the rival ones in these cases. The HAF algorithm yields the best results except for only one example when HA method is the most accurate.

The proposed methods are tested on 60 different planes, as it is shown in Table 3. The showed value is as follows: for every test plane homography is calculated by all the examined methods. Then the reprojection error of the homography computed by OpenCV is labeled with 100%. Other values in the table such as 66% related to HAF, means that the ratio of the average reprojection errors of HAF and OpenCV is 0.66.

Table 2: Reprojection errors (px) for sequences 'Model House' and 'Library'

	Model House		Library	
	Wall	Ground	Wall	Roof
OpenCV RSC	1.554	2.750	1.422	1.693
3PT RSC	1.400	1.569	1.513	1.399
HAF RSC	0.864	1.635	1.317	1.320
HA RSC	0.759	1.736	1.338	1.422
RHE Avg.	<i>2.911</i>	<i>4.819</i>	<i>7.889</i>	<i>2.445</i>
RHE Min.	<i>0.780</i>	<i>2.378</i>	<i>1.384</i>	<i>1.514</i>

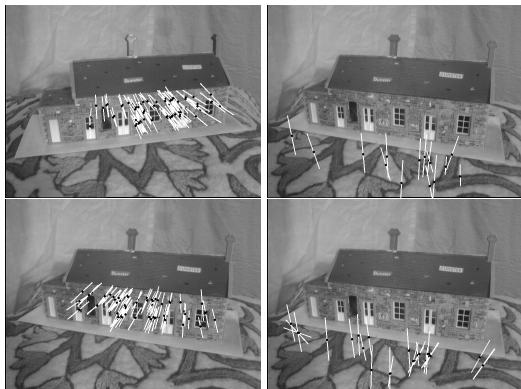


Figure 5: Reconstructed surface normals using RHE algorithm on sequence 'Model House'. Left: reconstructed wall Right: reconstructed floor. Top: first image. Bottom: second image.

4.3 Processing times

The processing time of each method is discussed here. HA and HAF methods are based on the solution of a homogeneous, and inhomogeneous linear system of equations, respectively. These systems consists of 6

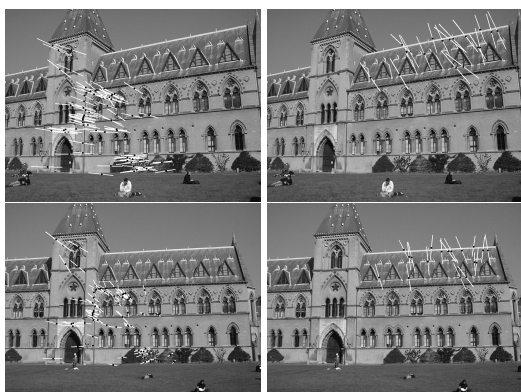


Figure 6: Reconstructed surface normals using RHE algorithm on sequence 'Library'. Left: reconstructed wall Right: reconstructed roof. Top: first image. Bottom: second image.

Table 3: Error percentage compared to OpenCV on 60 different planes.

OpenCV	3PT	HA	HAF	RHE Avg.	RHE Min.
100%	79%	67%	66%	119%	57%

and 4 equations per point pair. Therefore, HA is a bit slower than DLT, however, not significantly. HAF is as fast as DLT since the equation number per point is equal.

Even though, RHE is a numerical optimization in a 1-DoF search space, our implementation is not applicable to online tasks since its processing time is around half a second. However, it could be parallelised on GPU straightforwardly.

5 CONCLUSION

Novel homography estimator methods (HA and HAF) have been proposed here that can estimate the homography if the affine transformations are known between the surrounding regions of the corresponding point pairs. We have also proposed an algorithm to estimate the homography based on both point correspondences and photo-consistency.

HA method does not need the knowledge of epipolar geometry, however, it gives better results than the standard homography estimation techniques in most of the situations. As a minimal problem, it is computable from only two point correspondences and the related affine transformations. The HAF algorithm requires the knowledge of the fundamental matrix, and at least one point correspondence and the related affine transformation have to be known to calculate the homography. It is usually the most efficient method. Their RANSAC variants are recommended to use for contaminated input data, because affine transformations are significantly more sensitive to noise than point correspondences.

It is proven that affine-invariance is equivalent to perspective-invariance in the case of known fundamental matrix. We think it is a significant contribution to the theory of 3D stereo vision.

The novelty of the proposed RHE algorithm is to reduce homography estimation to a one-dimensional search in a half unit circle when both point correspondences and camera parameters are known. The similarity function for the minimization problem is based on photo-consistency.

The synthetic and real tests have shown that all the proposed methods (HA and HAF) give more accurate results and use similar amount of resources as

the state-of-the-art point correspondence-based techniques. Therefore the novel and standard algorithms can be easily replaced for each other. RHE algorithm also gives appropriate results using only two corresponding point pairs. Moreover, RHE gives accurate estimation in offline applications by repeating the optimization for many possible pairings. Then the point pair which supplies the best homography by RHE are usually more accurate than the results of all the other methods. It is important to note that if many point correspondences (hundreds of points) are given from the observed plane, the original point-based homography estimation methods give nearly the same result as the proposed ones.

Acknowledgement. The research was partially supported by the Hungarian Scientific Research Fund (OTKA No. 106374).

APPENDIX

Affine Transformation from Homography

The affine parameters can be obtained from homography between corresponding patches in stereo image pairs. Let us assume that the homography H is given. Then the correspondence between the coordinates in the first (u and v) and second (u' and v') images is written as

$$u' = \frac{h_1^T [u, v, 1]^T}{h_3^T [u, v, 1]^T} \quad v' = \frac{h_2^T [u, v, 1]^T}{h_3^T [u, v, 1]^T}$$

where the 3×3 homography matrix H is written as

$$H = \begin{bmatrix} h_1^T \\ h_2^T \\ h_3^T \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}$$

The affine parameters come from the partial derivatives of the perspective plane-plane transformation. The top left element a_{11} of affine transformation matrix is as follows:

$$a_{11} = \frac{\partial u'}{\partial u} = \frac{h_{11}h_3^T [u, v, 1]^T - h_{31}h_1^T [u, v, 1]^T}{(h_3^T [u, v, 1]^T)^2} = \frac{h_{11} - h_{31}u'}{s} \quad (9)$$

where

$$s = h_3^T [u, v, 1]^T \quad (10)$$

The other components of affine matrix are obtained similarly

$$a_{12} = \frac{\partial u'}{\partial v} = \frac{h_{12} - h_{32}u'}{s} \quad (11)$$

$$a_{21} = \frac{\partial v'}{\partial u} = \frac{h_{21} - h_{31}v'}{s} \quad (12)$$

$$a_{22} = \frac{\partial v'}{\partial v} = \frac{h_{22} - h_{32}v'}{s}. \quad (13)$$

Normalization of Affine Transformation

Given corresponding point pairs $x^{(1)}$ and $x^{(2)}$, the goal is to determine the related affine transformations if the points are normalized as $x^{(2)} = T_2 x^{(1)}$ and $x^{(1)} = T_1 x^{(1)}$. The normalization is the concatenation of a scale and a translation. Therefore, the transformation matrices can be written as

$$T_1 = \begin{bmatrix} s_x^{(1)} & 0 & t_x^{(1)} \\ 0 & s_y^{(1)} & t_y^{(1)} \\ 0 & 0 & 1 \end{bmatrix}, T_2 = \begin{bmatrix} s_x^{(2)} & 0 & t_x^{(2)} \\ 0 & s_y^{(2)} & t_y^{(2)} \\ 0 & 0 & 1 \end{bmatrix}. \quad (14)$$

For an arbitrary 2D point $x^{(i)} = [u^{(i)}, v^{(i)}]$ on the i -th image, the transformed coordinates can be written as

$$x'^{(i)} = \begin{bmatrix} s_x^{(i)} & 0 & t_x^{(i)} \\ 0 & s_y^{(i)} & t_y^{(i)} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u^{(i)} \\ v^{(i)} \\ 1 \end{bmatrix} = \begin{bmatrix} s_x^{(i)} u^{(i)} + t_x^{(i)} \\ s_y^{(i)} v^{(i)} + t_y^{(i)} \\ 1 \end{bmatrix}.$$

If the homography of a plane is denoted by H using the original coordinates, it connects the coordinates on the first and second image as $x_2 \sim Hx_1$. If the normalized coordinates are used, the relationship modifies as $T_2^{-1}x^{(2)} \sim HT_1^{-1}x^{(1)}$.

Therefore, the homography using the normalized coordinates is $H' = T_2HT_1^{-1}$. The deviations are written in Eqs. 15 – 18.

For the sake of simplicity, we do not determine the last two elements of the first row as they do not affect the affine transformation. They are denoted by stars (*'). The elements of the affine transformation are written in Eqs. 9 – 13. The normalized scale s' is written as

$$s' = \frac{1}{s_x^{(1)}} h_{31} \left(u^{(1)} - t_x^{(1)} \right) + \frac{1}{s_y^{(1)}} h_{32} \left(v^{(1)} - t_y^{(1)} \right) + h_{33} = u^{(1)} h_{31} + v^{(1)} h_{32} + h_{33} = s$$

Therefore, the normalization does not modify the scale as it is expected. Now, the numerator for the first

$$T_1^{-1} = \begin{bmatrix} 1/s_x^{(1)} & 0 & -t_x^{(1)}/s_x^{(1)} \\ 0 & 1/s_y^{(1)} & -t_y^{(1)}/s_y^{(1)} \\ 0 & 0 & 1 \end{bmatrix} \quad (15)$$

$$H' = T_2 H T_1^{-1} = \begin{bmatrix} s_x^{(2)} & 0 & t_x^{(2)} \\ 0 & s_y^{(2)} & t_y^{(2)} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} 1/s_x^{(1)} & 0 & -t_x^{(1)}/s_x^{(1)} \\ 0 & 1/s_y^{(1)} & -t_y^{(1)}/s_y^{(1)} \\ 0 & 0 & 1 \end{bmatrix} \quad (16)$$

$$H' = \begin{bmatrix} s_x^{(2)} h_{11} + t_x^{(2)} h_{31} & s_x^{(2)} h_{12} + t_x^{(2)} h_{32} & s_x^{(2)} h_{13} + t_x^{(2)} h_{33} \\ s_y^{(2)} h_{21} + t_y^{(2)} h_{31} & s_y^{(2)} h_{22} + t_y^{(2)} h_{32} & s_y^{(2)} h_{23} + t_y^{(2)} h_{33} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} 1/s_x^{(1)} & 0 & -t_x^{(1)}/s_x^{(1)} \\ 0 & 1/s_y^{(1)} & -t_y^{(1)}/s_y^{(1)} \\ 0 & 0 & 1 \end{bmatrix} \quad (17)$$

$$H' = \begin{bmatrix} \frac{s_x^{(2)}}{s_x^{(1)}} h_{11} + \frac{t_x^{(2)}}{s_x^{(1)}} h_{31} & \frac{s_x^{(2)}}{s_x^{(1)}} h_{12} + \frac{t_x^{(2)}}{s_x^{(1)}} h_{32} & * \\ \frac{s_y^{(2)}}{s_y^{(1)}} h_{21} + \frac{t_y^{(2)}}{s_y^{(1)}} h_{31} & \frac{s_y^{(2)}}{s_y^{(1)}} h_{22} + \frac{t_y^{(2)}}{s_y^{(1)}} h_{32} & * \\ \frac{1}{s_x^{(1)}} h_{31} & \frac{1}{s_y^{(1)}} h_{32} & -h_{31} t_x^{(1)}/s_x^{(1)} - h_{32} t_y^{(1)}/s_y^{(1)} + h_{33} \end{bmatrix} \quad (18)$$

affine transformation can be expressed as follows:

$$\begin{aligned} h'_{11} - h'_{31} u^{(2)} &= \\ \frac{s_x^{(2)}}{s_x^{(1)}} h_{11} + \frac{t_x^{(2)}}{s_x^{(1)}} h_{31} - \frac{1}{s_x^{(1)}} h_{31} (s_x^{(2)} u^{(2)} + t_x^{(2)}) &= \\ \frac{s_x^{(2)}}{s_x^{(1)}} h_{11} - \frac{s_x^{(2)}}{s_x^{(1)}} u^{(2)} h_{31} & \end{aligned}$$

The other three components of the transformation can be computed similarly:

$$\begin{aligned} h'_{12} - h'_{32} u^{(2)} &= \frac{s_x^{(2)}}{s_x^{(1)}} h_{12} - \frac{s_x^{(2)}}{s_x^{(1)}} u^{(2)} h_{32} \\ h'_{21} - h'_{31} v^{(2)} &= \frac{s_y^{(2)}}{s_y^{(1)}} h_{21} - \frac{s_y^{(2)}}{s_y^{(1)}} v^{(2)} h_{31} \\ h'_{22} - h'_{32} v^{(2)} &= \frac{s_y^{(2)}}{s_y^{(1)}} h_{22} - \frac{s_y^{(2)}}{s_y^{(1)}} v^{(2)} h_{32} \end{aligned}$$

By rearranging the equations the following formulas are given:

$$\begin{aligned} (h_{31} u^{(1)} + h_{32} v^{(2)} + h_{33}) a'_{11} &= \frac{s_x^{(2)}}{s_x^{(1)}} h_{11} - \frac{s_x^{(2)}}{s_x^{(1)}} u^{(2)} h_{31} \quad (19) \\ (h_{31} u^{(1)} + h_{32} v^{(1)} + h_{33}) a'_{12} &= \frac{s_x^{(2)}}{s_x^{(1)}} h_{12} - \frac{s_x^{(2)}}{s_x^{(1)}} u^{(2)} h_{32} \\ (h_{31} u^{(1)} + h_{32} v^{(1)} + h_{33}) a'_{21} &= \frac{s_y^{(2)}}{s_y^{(1)}} h_{21} - \frac{s_y^{(2)}}{s_y^{(1)}} v^{(2)} h_{31} \\ (h_{31} u^{(1)} + h_{32} v^{(1)} + h_{33}) a'_{22} &= \frac{s_y^{(2)}}{s_y^{(1)}} h_{22} - \frac{s_y^{(2)}}{s_y^{(1)}} v^{(2)} h_{32} \end{aligned}$$

These equations are linear w.r.t. the elements of the homography. Therefore, these formulas compose a homogeneous, linear system of equations. In order to apply affine normalization to the proposed methods, the equations refer to the affine transformations have to be replaced in the coefficient matrix of each method. For HAF a few modifications are required beforehand. Formulas, which describe the connection to the fundamental matrix (Eq. 4) have to substituted into Eq. 19. The given equations are inhomogeneous due to the elements of matrix F . After a few modifications these can also be substituted into the coefficient matrix of HAF (Eq. 8).

REFERENCES

- Agarwal, A., Jawahar, C., and Narayanan, P. (2005). A Survey of Planar Homography Estimation Techniques. Technical report, IIT-Hyderabad.
- Agarwal, S., Furukawa, Y., Snavely, N., Simon, I., Curless, B., Seitz, S. M., and Szeliski, R. (2011). Building rome in a day. *Commun. ACM*, 54(10):105–112.
- B., D. (1934). Sur la sphere vide. *Izvestia Akademii Nauk SSSR, Otdelenie Matematicheskikh i Estestvennykh Nauk*, 7:793–800.
- Barath, D., Molnar, J., and Hajder, L. (2015). Optimal Surface Normal from Affine Transformation. In *VISAPP 2015*, pages 305–316.
- Björck, Å. (1996). *Numerical Methods for Least Squares Problems*. Siam.
- Bódis-Szomorú, A., Riemenschneider, H., and Gool, L. V. (2014). Fast, approximate piecewise-planar modeling based on sparse structure-from-motion and superpixels. In *IEEE Conference on Computer Vision and Pattern Recognition*.
- Faugeras, O. and Lustman, F. (1988). Motion and struc-

- ture from motion in a piecewise planar environment. Technical Report RR-0856, INRIA.
- Faugeras, O. D. and Papadopoulos, T. (1998). A Nonlinear Method for Estimating the Projective Geometry of Three Views. In *ICCV*, pages 477–484.
- Fischler, M. and Bolles, R. (1981). RANdom SAMpling Consensus: a paradigm for model fitting with application to image analysis and automated cartography. *Commun. Assoc. Comp. Mach.*, 24:358–367.
- Furukawa, Y. and Ponce, J. (2010). Accurate, dense, and robust multi-view stereopsis. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 32(8):1362–1376.
- Habbecke, M. and Kobbelt, L. (2006). Iterative multi-view plane fitting. In *Proceeding of Vision, Modelling, and Visualization*, pages 73–80.
- Hartley, R. I. and Sturm, P. (1997). Triangulation. *Computer Vision and Image Understanding: CVIU*, 68(2):146–157.
- Hartley, R. I. and Zisserman, A. (2003). *Multiple View Geometry in Computer Vision*. Cambridge University Press.
- He, L. (2012). *Deeper Understanding on Solution Ambiguity in Estimating 3D Motion Parameters by Homography Decomposition and its Improvement*. PhD thesis, University of Fukui.
- Kanatani, K. (1998). Optimal homography computation with a reliability measure. In *Proceedings of IAPR Workshop on Machine Vision Applications, MVA*, pages 426–429.
- Kannala, J., Salo, M., and Heikkilä, J. (2006). Algorithms for computing a planar homography from conics in correspondence. In *Proceedings of the British Machine Vision Conference*.
- Kruger, S. and Calway, A. (1998). Image registration using multiresolution frequency domain correlation. In *Proceedings of the British Machine Vision Conference*.
- Kumar, M. P., Goyal, S., Kuthirummal, S., Jawahar, C. V., and Narayanan, P. J. (2004). Discrete contours in multiple views: approximation and recognition. *Image and Vision Computing*, 22(14):1229–1239.
- Lee, D.-T. and Schachter, B. J. (1980). Two algorithms for constructing a delaunay triangulation. *International Journal of Computer & Information Sciences*, 9(3):219–242.
- Malis, E. and Vargas, M. (2007). Deeper understanding of the homography decomposition for vision-based control. Technical Report RR-6303, INRIA.
- Marquardt, D. (1963). An algorithm for least-squares estimation of nonlinear parameters. *SIAM J. Appl. Math.*, 11:431–441.
- Mikolajczyk, K. and Schmid, C. (2004). Scale & affine invariant interest point detectors. *International journal of computer vision*, 60(1):63–86.
- Molnár, J. and Chetverikov, D. (2014). Quadratic transformation for planar mapping of implicit surfaces. *Journal of Mathematical Imaging and Vision*, 48:176–184.
- Molnár, J., Huang, R., and Kato, Z. (2014). 3d reconstruction of planar surface patches: A direct solution. *ACCV Big Data in 3D Vision Workshop*.
- Morel, J.-M. and Yu, G. (2009). Asift: A new framework for fully affine invariant image comparison. *SIAM Journal on Imaging Sciences*, 2(2):438–469.
- Mudigonda, P. K., Kumar, P., Jawahar, M. C. V., and Narayanan, P. J. (2004). Geometric structure computation from conics. In *In ICVGIP*, pages 9–14.
- Murino, V., Castellani, U., Etrari, A., and Fusiello, A. (2002). Registration of very time-distant aerial images. In *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, volume III, pages 989–992. IEEE Signal Processing Society.
- Musialski, P., Wonka, P., Aliaga, D. G., Wimmer, M., van Gool, L., and Purgathofer, W. (2012). A survey of urban reconstruction. In *EUROGRAPHICS 2012 State of the Art Reports*, pages 1–28.
- Pollefeys, M., Nistér, D., Frahm, J. M., Akbarzadeh, A., Mordohai, P., Clipp, B., Engels, C., Gallup, D., Kim, S. J., Merrell, P., Salmi, C., Sinha, S., Talton, B., Wang, L., Yang, Q., Stewénius, H., Yang, R., Welch, G., and Towles, H. (2008). Detailed real-time urban 3d reconstruction from video. *Int. Journal Comput. Vision*, 78(2-3):143–167.
- Semple, J. and Kneebone, G. (1952). *Algebraic Projective Geometry*. Oxford University Press.
- Sonka, M., Hlavac, V., and Boyle, R. (2007). *Image Processing, Analysis, and Machine Vision*. CENGAGE-Engineering, third edition edition.
- Tanács, A., Majdik, A., Hajder, L., Molnár, J., Sánta, Z., and Kato, Z. (2014). Collaborative mobile 3d reconstruction of urban scenes. In *Computer Vision - ACCV 2014 Workshops - Singapore, Singapore, November 1-2, 2014, Revised Selected Papers, Part III*, pages 486–501.
- Toldo, R. and Fusiello, A. (2010). Real-time incremental j-linkage for robust multiple structures estimation. In *International Symposium on 3D Data Processing, Visualization and Transmission (3DPVT)*, volume 1, page 6.
- Vu, H.-H., Labatut, P., Pons, J.-P., and Keriven, R. (2012). High accuracy and visibility-consistent dense multi-view stereo. *IEEE Trans. Pattern Anal. Mach. Intell.*, 34(5):889–901.
- Z. Megyesi, G. and D. Chetverikov (2006). Dense 3d reconstruction from images by normal aided matching. *Machine Graphics and Vision*, 15:3–28.
- Zhang, Z. (2000). A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334.