sensors

MDPI

Article

Homography Ranking Based on Multiple Groups of Point Correspondences

Milan Ondrašovič * and Peter Tarábek

Faculty of Management Science and Informatics, University of Žilina, Univerzitná 8215/1, 010 26 Žilina, Slovakia; peter.tarabek@fri.uniza.sk

* Correspondence: milan.ondrasovic@fri.uniza.sk

Abstract: Homography mapping is often exploited to remove perspective distortion in images and can be estimated using point correspondences of a known object (marker). We focus on scenarios with multiple markers placed on the same plane if their relative positions in the world are unknown, causing an indeterminate point correspondence. Existing approaches may only estimate an isolated homography for each marker and cannot determine which homography achieves the best reprojection over the entire image. We thus propose a method to rank isolated homographies obtained from multiple distinct markers to select the best homography. This method extends existing approaches in the post-processing stage, provided that the point correspondences are available and that the markers differ only by similarity transformation after rectification. We demonstrate the robustness of our method using a synthetic dataset and show an approximately 60% relative improvement over the random selection strategy based on the homography estimation from the OpenCV library.

Keywords: homography matrix; many-to-one point correspondence; perspective distortion; ranking method; bird's-eye view

check for updates

Citation: Ondrašovič, M.; Tarábek, P. Homography Ranking Based on Multiple Groups of Point Correspondences. *Sensors* **2021**, *21*, 5752. https://doi.org/10.3390/s21175752

Academic Editor: Andreas Savakis

Received: 8 July 2021
Accepted: 24 August 2021
Published: 26 August 2021

1. Introduction

Homography is a perspective projection of a plane from one camera view into a different camera view. The perspective projection maps points from a 3D world onto a 2D image plane along lines that emanate from a single point [1,2]. This projection is performed by a 3×3 invertible transformation matrix called the homography matrix (or just homography) with eight degrees of freedom (DoF). In the pinhole camera model, any two images of the same planar surface are related to each other by the homography [3,4]. Homography is commonly used for the rectification of text document images by generating a fronto-parallel view [5,6], image stitching [7,8], video stabilization [9], extracting metric information from 2D images [10], and pose estimation [11] and for various traffic-related applications, e.g., ground-plane detection [12] and bird's-eye view projection [13].

Homography estimation is essential for image registration, i.e., a process of image matching and transformation of two or more different images [14]. It can be addressed either on the pixel or feature levels. In our work, we focus on feature-based approaches that utilize only a subset of pixels. A common approach to estimating the homography is to use a set of at least four 2D point correspondences [4]. We refer to the points used for establishing the 2D point correspondences as keypoints. These keypoints may belong to a marker, which is an object with a known shape that is either naturally occurring or artificially positioned in the scene. A regular pattern such as a chessboard is usually utilized [15]. A single marker is identified in the image by multiple independent keypoints that have a direct correspondence to its real shape, thus making a group of point correspondences. However, these correspondences are often noisy and they can introduce errors in the homography estimation. Although four keypoints are satisfactory, often a greater number of keypoints is used, allowing us to use optimization to minimize a suitable cost function [16,17]. Then, outlier removal becomes an important step, and algorithms such as RANSAC [18] are usually employed [16].

Assume the presence of a sole marker at the scene. Even though the marker is distorted under perspective, the knowledge of its real shape makes it possible to compute the homography. When multiple copies of the same marker are visible but their positions in the world are unknown, the knowledge of the shape is not enough to incorporate all of the keypoints in the estimation. In the absence of position information, existing approaches for homography estimation based on point correspondences fail because the projection has to preserve the proportional positions. Thus, estimating the homography without knowing the ground-truth layout of the keypoints up to an arbitrary scale does not guarantee the correct result. Under the aforementioned constraints, existing methods can only generate an isolated homography for each marker based on the one-to-one point correspondence (see Figure 1). Each homography may be affected by different sources of noise, e.g., low resolution, blur, or keypoint detection. Thus, the outcome of rectification may vary. Additionally, in many practical applications, a single marker usually covers a small portion of the image, which increases susceptibility to noise. The trivial solution would be to use a bigger marker that covers the majority of the estimated plane in the image. However, this solution is often impractical. Furthermore, it is not possible to simply "merge" multiple isolated homographies together.

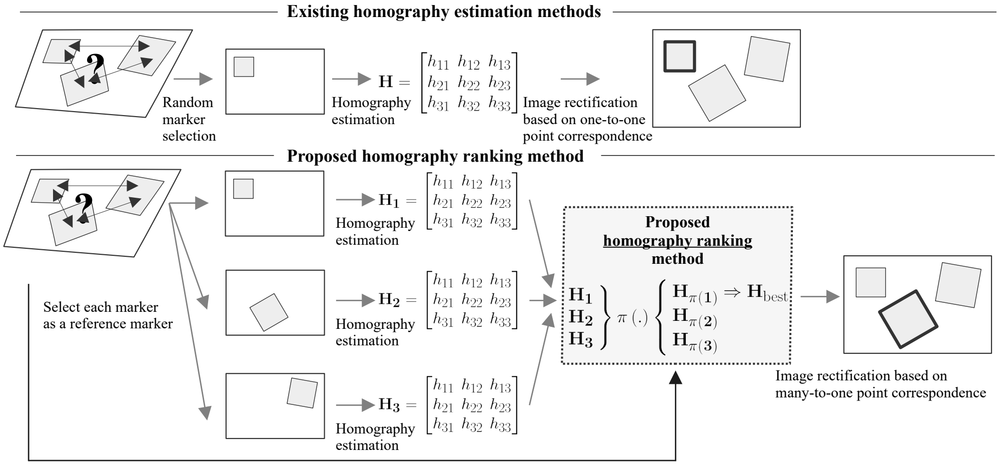
Existing homography estimation methods
Random marker selection
H = $\begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}$
Homography estimation
Image rectification based on one-to-one point correspondence
Proposed homography ranking method
Select each marker as a reference marker
$\mathbf{H_1}$ =
$\mathbf{H_2}$ =
$\mathbf{H_3}$ =
Homography estimation
Proposed homography ranking method
$\mathbf{H_1}$ $\mathbf{H_2}$ $\mathbf{H_3}$ $\pi(.)$
$\mathbf{H}_{\pi(1)} \Rightarrow \mathbf{H}_{\text{best}}$
$\mathbf{H}_{\pi(2)}$
$\mathbf{H}_{\pi(3)}$
Image rectification based on many-to-one point correspondence

Figure 1. The difference between existing homography estimation methods and the proposed homography ranking method. In the presence of multiple markers without information about their relative positions in the world, existing approaches can only estimate isolated homographies without the ability to select the best one. Our method extends existing approaches by exploiting multiple markers to rank the isolated homographies.

In this paper, we focus on exploiting information from multiple markers, i.e., multiple groups of point correspondences. We assume that the markers are placed on the same plane in the world over which we want to acquire the bird's-eye view. We thus propose a homography ranking method that can incorporate information about multiple markers to select homography with the potentially minimal reprojection error. Therefore, the problem lies in determining which homography potentially achieves the best reprojection accuracy over the entire image. The proposed ranking method allows us to systematically select the best homography according to our score function when multiple choices are available. The outcome of our algorithm may serve as a recommendation for homography selection. We emphasize that we are not concerned with homography estimation itself. We only require point correspondences. Our algorithm can be implemented as an extension to existing approaches to sort a set of already estimated homographies according to their potential reprojection accuracy. We sidestep the need for the position information by constraining the markers' shape. We assume that the markers in the world, while placed on the same plane, differ only in translation, rotation, and uniform scaling, i.e., a similarity transformation exists between them when viewed without perspective (see Section 3.1). The proposed method ranks homographies using our score function that computes a single value for each transformation matrix (see Section 3.2). The score value is used as a proxy to measure the reprojection "quality" of a specific homography over the whole image.

This work was motivated by a real-world application of generating a bird's-eye view over a road from a video recording when we could not use a large marker to cover a sufficient portion of the road. Homography estimation based on a single small marker was inaccurate. Therefore, we tried to use multiple small markers and to measure their relative positions. However, their position measurements were highly noisy at best. Thus, the proposed method was used instead. Our method can also be adopted in a situation when a marker placed at various positions on the same planar surface is visible at different frames using a static camera. Stacking the frames onto each other yields a view with multiple markers.

Due to the exploitation of similarity transformations, the limitation of our approach is that it can only handle the projection from a distorted to the undistorted view of the target plane, not between various projective perspectives of the same plane. Therefore, it serves the removal of perspective distortion.

The experiments showed that the proposed method could systematically improve reprojection error by selecting the best homography according to our score function. We quantified the relative improvement in terms of reprojection accuracy ratio between the systematic homography selection and the baseline random selection. We used a random selection because the existing methods could not compare the "quality" of individual homographies and were therefore left with a random selection or some subjective rules. In practice, random selection would often be replaced by an educated guess. Without the loss of generality, the homographies in our tests were estimated using the implementation from the OpenCV [19] library. Thus, the main contribution of our work is as follows:

- The proposed method ranks (sorts) multiple homographies corresponding to individual markers placed on the same plane to select the "best" homography for rectification. Our method handles the absence of position information between markers in the world and builds on top of many-to-one point correspondences. The algorithm is an extension of existing methods since it works with already estimated homography matrices and does not alter them. This easy-to-implement extension is efficient, with a quadratic algorithmic complexity in the number of markers, which is usually very low.

The rest of the paper is organized as follows. The upcoming Section 2 contains an overview of related work. Then, in Section 3, we describe our proposed method. Section 4 is devoted to experiments and their evaluation. We summarize our conclusions in Section 5.

2. Related Work

To the best of our knowledge, there is no work related to the same narrow use case of homography transformation as what we deal with. In principle, our method can extend any homography estimation approach that satisfies the requirements. Therefore, in this section, we dissect various ways to estimate homographies and other works in which techniques that intersect with ours were employed.

2.1. Single Homography Estimation

Homography can be estimated using at least four point correspondences [3]. However, for this task, linear methods are sensitive to noise even if there are no outliers. To this end, many pre-processing steps have been developed. For example, normalizing each point set by translating the center of mass to the origin and by scaling appropriately [4]. Our goal is to tackle situations where point correspondences contain noise and thus outliers [20]. We mention markers as a demonstration of point correspondence. A paper that builds on fiducial markers and further homography refining is [21]. The authors discuss square and circular markers and propose a method to make extra adjustments to the initial homography estimate using point correspondences. Although we only focus on point correspondences, it is not the only way to identify a relationship between the observed marker and its ground-truth shape. For instance, circular markers pose new possibilities (e.g., the exploitation of vanishing lines) as well as challenges (e.g., ambiguity) for homography estimation. For a more detailed discussion, see [11].

If the system of equations formed by the point correspondences is overdetermined, then methods such as RANSAC [18] are used to separate inliers from outliers. Zhu et al. [22] developed an efficient algorithm to estimate the homography based on order-preserving constraints. In specific use cases, it is faster than RANSAC. As shown in [17], optimization-based approaches perform well with a large number of outliers.

Jawahar et al. [23] used object contours instead of point correspondences for homography estimation. Their algorithm started from affine transformation and iteratively advanced towards homography. We incorporate affine (similarity) transformations in our computations as well, but no iterative refining is involved. Chen et al. [24] proposed an iterative approach for homography estimation using point correspondences. Their contribution was to adopt a more reliable nonlinear geometric error rather than just an algebraic error. A thorough discussion of various computations of errors for homographies, including the geometric error, can be found in [25]. We employed the geometric error, too (see Section 3.2). Likewise, Li et al. [26] measured the reprojection error using the l_2-norm and showed that it is suitable for homography estimation.

Our core idea of assessing the quality of a homography matrix in terms of removing perspective distortion consists of measuring how accurately multiple objects with known shapes align with the expected shape after rectification. Song et al. [27] proposed a homography matrix evaluation method based on a geometric approach to increase the accuracy of aerial image matching. They assessed the transformation accuracy of a given homography by examining the shape of a transformed quadrangle. They matched the reference aerial image with the sensed image by iteratively refining the homography governed by the evaluation procedure. We do not have a reference image since we build on top of the reference object instead. One way to identify degenerate homographies is to compute the determinant or the condition number of the matrix [28]. Thus, the matrices can be assessed in terms of their "quality", which is the purpose of our work. We exploited this property in one of our experiments concerning a homography optimization procedure we developed. See Appendix A for more details.

2.2. Multiple Homography Estimation

Bose et al. [29] presented a technique for a full affine and metric rectification of the ground plane by tracking moving objects. Their work is similar to ours in the exploitation of multiple instances of the same object at various places and the measurement of its properties. They estimated vanishing points based on non-parallel object trajectories to obtain the projective transformation and then used other geometric clues to deduce the affinity. On the other hand, several works use multiple planes to estimate a homography or to directly utilize multiple homographies.

A common strategy is to introduce additional constraints. In our work, we focus on one plane only with multiple already estimated homographies. Our constraints are related to similarity transformations. Taking advantage of multiple views of several planar surfaces may improve the single homography estimation since isolated plane homographies are compatible unless noise is present. In such a case, denoising constraints have to be introduced [30]. The work of [31] inspired [32] to tackle planar mapping and tracking by exploiting multiple frames and plane-induced homographies between them. Their system used nonlinear optimization. Chojnacki et al. [33] estimated multiple homographies linked together by consistency constraints. They estimated a set of homography matrices induced by multiple planes in the 3D scene between two views. Ruiz et al. [31] remarked that geometric constraints among multiple views could be used to recover a projective transformation. They proposed a simple 2D optimization method for the metric rectification of a single plane from several perspective images. In their setting, the camera(s) moved and a single object of interest was stationary. We either utilize a single moving object or multiple similar stationary objects in the scene under the assumption of being placed on the same planar surface.

Park et al. [34] proposed a panorama stitching method based on multiple frames using homography under the assumption of a static camera. The goal was to suppress the incorrect feature point extraction caused by time-varying noise to find the proper alignment parameters by estimating multiple homographies during a predetermined time intervals. To evaluate the accuracy, they employed metrics based on alignment distortion measurement. Cui [35] also highlighted the importance of homography to the segmentation of moving objects. Their proposed method allowed for using static and moving cameras by exploiting constraints based on multiple overlapped homographies.

Fraundorfer et al. [36] proposed a method that could recover scene planes of arbitrary position and orientation using multiple homographies and point correspondences. They employed iterative refining of the plane-induced homographies. We adopted a similar approach for evaluation. The authors also created a synthetic dataset with artificial fiducial markers in the scene. DeTone et al. [37] and Zhu et al. [22] adopted similar approaches to generate the synthetic homography-related dataset, too.

2.3. Deep Learning-Based Approaches

Some recent approaches make use of deep learning. In [37], they trained a neural network in a supervised manner to estimate the homography, whereas in [38], the homography was estimated from context using unsupervised learning. Deep learning favors use cases that pose a challenge for traditional approaches, e.g., dynamic scenes. Le et al. [39] proposed a deep learning-based approach to identify dynamic content in images and to estimate homography from coarse to fine using a multi-scale neural network trained in a multi-task fashion. They avoided iterative processes such as RANSAC. They also developed a suitable dataset as a response to the lack of available homography-related training data. Considering content awareness, the work of Zhao et al. [40] showed that deep learning boosts image stitching if the loss function considers image content. Homography estimation using deep learning was exploited in [41] to improve foreground segmentation. The majority of deep learning-based approaches still rely on four point correspondences. However, still, emerging works that estimate the homography directly instead of explicitly harnessing point correspondences show promising results, for instance [42].

3. Proposed Method

3.1. Preliminaries

A marker is an object with a known, easy-to-detect shape. This object either naturally occurs or is artificially placed on the planar surface of the scene we want to produce a bird's-eye view for, i.e., to remove perspective distortion. The marker contains keypoints, a set of distinct, independent, visual feature points, e.g., corners. Thechosen keypoints visible in the perspectively deformed image are called the warped keypoints. The set of the rectified keypoints in the desired image (not subjected to perspective distortion) is produced from the warped keypoints using the homography projection. The point correspondence is a relationship between the warped and the target keypoints, and it is used for homography estimation. Ideally, the rectified keypoints match the target keypoints. See Figure 2 for details.

The goal of homography estimation is to find the 3 × 3 homography matrix

$$\mathbf{H} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \tag{1}$$

with 8 DoF [4]. A single vector $\mathbf{u}^T = \left[u_x, u_y, 1\right]$, representing a warped keypoint in homogeneous coordinates, is mapped onto the rectified keypoint $\tilde{\mathbf{u}}^T = \left[\tilde{u}_x, \tilde{u}_y, 1\right]$ by the homography $\mathbf{H}$ using the transformation $s\tilde{\mathbf{u}} \approx \mathbf{H}\mathbf{u}$, with s being the scale factor.

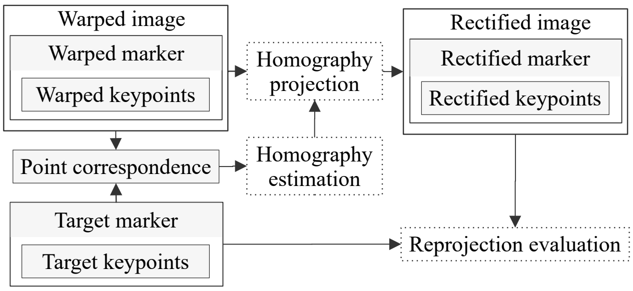

Figure 2. Visualization of relationships in our established terminology. The diagram also shows the hierarchical dependence of individual terms. Dotted elements represent processes with arrows denoting their input and output.

Without stating otherwise, a similarity transformation denotes a limited affine transformation with 4 DoF consisting of translation, rotation, and uniform scaling (Equation (5)). Let $\mathcal{K}_1$ and $\mathcal{K}_2$ be sets of feature keypoints belonging to objects O_1 and O_2. We say that objects O_1 and O_2 are similar if there exists a similarity transformation ψ such that $\mathcal{K}_1 = \psi(\mathcal{K}_2)$ and $\mathcal{K}_2 = \psi^{-1}(\mathcal{K}_1)$. For example, O_1 and O_2 may be rectangles of different sizes but with an identical aspect ratio.

Let m be the number of markers and k be the number of keypoints of each marker. Each ith marker is described by a $3 \times k$ matrix $\mathbf{W}^{(i)}$ containing its warped keypoints as

$$\mathbf{W}^{(i)} = \begin{bmatrix} x_1^{(i)} & x_2^{(i)} & \dots & x_k^{(i)} \\ y_1^{(i)} & y_2^{(i)} & \dots & y_k^{(i)} \\ 1 & 1 & \dots & 1 \end{bmatrix}, i = 1, \dots, m. \tag{2}$$

The target keypoints are specified analogically by the $3 \times k$ matrix $\mathbf{T}$. Only one specification is sufficient due to many-to-one correspondence. The ordering of keypoints needs to match the warped keypoints defined above. Thus,

$$\mathbf{T} = \begin{bmatrix} \tilde{x}_1 & \tilde{x}_2 & \dots & \tilde{x}_k \\ \tilde{y}_1 & \tilde{y}_2 & \dots & \tilde{y}_k \\ 1 & 1 & \dots & 1 \end{bmatrix}, \tag{3}$$

with the point correspondence being

$$x_j^{(i)} \simeq \tilde{x}_j, y_j^{(i)} \simeq \tilde{y}_j, i = 1, \dots, m, j = 1, \dots, k. \tag{4}$$

3.2. Homography Ranking Algorithm

Our method utilizes multiple similar markers (see Figure 3). The input is point correspondences and homographies estimated for each marker. Each marker is selected exactly once as a reference marker. All remaining markers are in the role of auxiliary markers. The reference marker's homography is used to perform the perspective transformation to rectify all markers. To rank which reference markers' homography yields the best reprojection, we exploit auxiliary markers. Auxiliary markers are subsequently mapped onto the target marker using similarity transformations (Equation (5)). We then convert the transformed keypoints to homogeneous coordinates and measure the reprojection error as the mean Euclidean distance between the rectified and the target keypoints (7). The aim is to minimize this quantity. The optimal similarity matrices are just auxiliary and redundant after the algorithm ends.

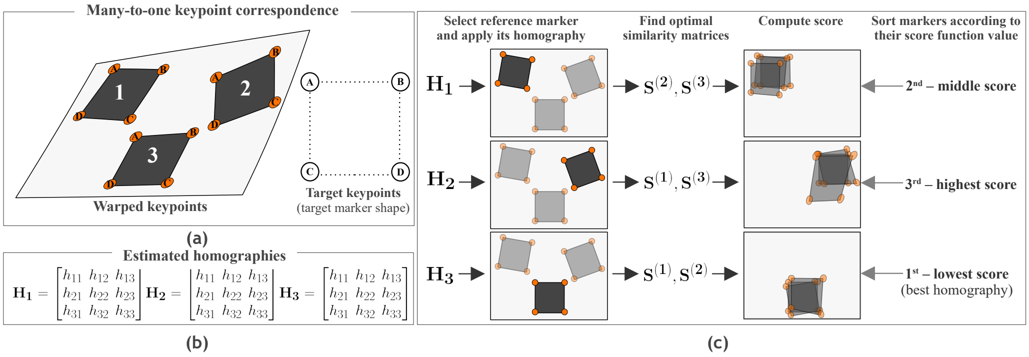
Many-to-one keypoint correspondence
1
2
3
A
B
C
D
Warped keypoints
Target keypoints
(target marker shape)
(a)
Estimated homographies
$H_1 = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}$
$H_2 = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}$
$H_3 = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}$
(b)
Select reference marker and apply its homography
Find optimal similarity matrices
Compute score
Sort markers according to their score function value
H_1
H_2
H_3
$S^{(2)}, S^{(3)}$
$S^{(1)}, S^{(3)}$
$S^{(1)}, S^{(2)}$
2nd – middle score
3rd – highest score
1st – lowest score
(best homography)
(c)

Figure 3. A system diagram describing the general idea behind our method. (**a**) The input consists of a many-to-one point correspondence specified by geometrically similar markers and information about the shape of the target marker. (**b**) We assume that the isolated homographies corresponding to each independent marker are provided on the input as well. (**c**) The algorithm processes each marker by applying its homography matrix to the image to produce a rectified image. Subsequently, it computes optimal similarity matrices corresponding to the auxiliary markers. The computation of the score function makes use of these transformations. The obtained score values then serve for comparison to rank (sort in ascending order) the homographies. The homography ranked first is considered the "best" candidate for the minimal reprojection error over the entire image.

Let r be the index of the reference marker. The 3×3 matrices describing similarity transformations are contained in a set $\mathcal{S} = \left\{ \mathbf{S}^{(i)} \mid i = 1, \ldots, m \right\}$, such that

$$\mathbf{S}^{(i)} = \begin{cases} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} & \text{if } i = r \\ \begin{bmatrix} \mathbf{R}^{(i)}_{2\times2} & \mathbf{T}^{(i)}_{2\times1} \\ \mathbf{0}_{1\times2} & 1 \end{bmatrix} & \text{if } i \neq r \end{cases}, \tag{5}$$

for $i = 1, \ldots, m$, where

$$\mathbf{R}^{(i)}_{2\times2} = \begin{bmatrix} s^{(i)} \cdot \cos\left(\theta^{(i)}\right) & -s^{(i)} \cdot \sin\left(\theta^{(i)}\right) \\ s^{(i)} \cdot \sin\left(\theta^{(i)}\right) & s^{(i)} \cdot \cos\left(\theta^{(i)}\right) \end{bmatrix}, \quad \mathbf{T}^{(i)}_{2\times1} = \begin{bmatrix} t^{(i)}_x \\ t^{(i)}_y \end{bmatrix}. \tag{6}$$

This transformation (except for the identity) consists of 4 DoF: single rotation angle $\theta^{(i)}$, two x and y translation coefficients $t^{(i)}_x$, $t^{(i)}_y$, and a scale coefficient $s^{(i)}$. A full affine transformation with 6 DoF would be responsible for horizontal and vertical scales, shear and rotation, and x and y offsets [43]. The application of homography that rectifies an image produces a frontal plane that is related to the ground-truth plane by similarity transformation [3,44]. Thus, we do not include the shear and we only support uniform scaling (see Appendix A.1 for explanation).

Since all of the markers share the same planar surface, any homography has to provide a valid perspective projection, but all perspective projections are subjected to different noise. Our goal is to quantify which homography estimation provides the best perspective projection for the whole plane in the image. To do so, we propose a score function based on the aforementioned constraints. The score function computes a score for individual homographies in conjunction with estimated similarity matrices corresponding to auxiliary markers as

$$\mathcal{F}(\mathbf{H}, \mathcal{S}) = \frac{1}{m} \sum_{i=1}^{m} \left\| h\left(\mathbf{S}^{(i)} \mathbf{H} \mathbf{W}^{(i)}\right) - \mathbf{T} \right\|_F, \tag{7}$$

where $\|\cdot\|_F$ denotes the Frobenius norm. The function $h(\cdot)$ converts points to homogeneous coordinates as

$$h\left(\begin{bmatrix} x_1 & x_2 & \dots & x_k \\ y_1 & y_2 & \dots & y_k \\ z_1 & z_2 & \dots & z_k \end{bmatrix}\right) = \begin{bmatrix} x_1/z_1 & x_2/z_2 & \dots & x_k/z_k \\ y_1/z_1 & y_2/z_2 & \dots & y_k/z_k \\ 1 & 1 & \dots & 1 \end{bmatrix}. \tag{8}$$

Now, we describe the proposed Algorithm 1 for homography ranking. Assume a set of warped markers described by warped keypoints and a single target marker described by target keypoints. These objects are linked by a many-to-one point correspondence. Additionally, assume that homographies have been estimated for each marker in isolation. Our algorithm ascendingly ranks the input set of all pairs $\left(\mathbf{W}^{(i)}, \mathbf{T}\right)$, $i = 1, \dots, m$, by how well each ith marker preserves the target shape of all the markers in the image after removing the perspective distortion. This objective is measured by the score function defined in Equation (7). The algorithm evaluates all markers as candidates for the reference marker. In each iteration, it computes optimal similarity matrices for the auxiliary markers in the rectified plane, i.e., after applying the perspective projection induced by the current homography. The aim is to find a homography with a minimal score. The algorithmic complexity is quadratic in the number of markers; thus, $\Theta(m(m-1) + m\log_2(m)) \simeq \Theta(m^2)$.

Algorithm 1 Homography ranking.

```
$\bar{\boldsymbol{H}} \leftarrow$ array$[m]$                                   ▷ homographies
$\boldsymbol{s} \leftarrow$ array$[m]$                                   ▷ scores
for $i \leftarrow 1, \dots, m$ do
    $\bar{\boldsymbol{H}}[i] \leftarrow$ HOMOGRAPHY($\mathbf{W}^{(i)}, \mathbf{T}$)            ▷ perspective
    $\bar{\mathbf{S}}^{(i)} \leftarrow \mathbf{I}_{3\times3}$
    $\bar{\mathcal{S}} \leftarrow \left\{\bar{\mathbf{S}}^{(i)}\right\}$                                ▷ similarity matrices
    for all $j : \{1, \dots, m\} - \{i\}$ do
        $\bar{\mathbf{S}}^{(j)} \leftarrow$ SIMILARITY($\bar{\boldsymbol{H}}[i] \cdot \mathbf{W}^{(j)}, \mathbf{T}$)
        $\bar{\mathcal{S}} \leftarrow \bar{\mathcal{S}} \cup \bar{\mathbf{S}}^{(j)}$
    end for
    $\boldsymbol{s}[i] \leftarrow \mathcal{F}(\bar{\boldsymbol{H}}[i], \bar{\mathcal{S}})$                          ▷ Equation (7)
end for
$\boldsymbol{\omega} \leftarrow$ ARGSORT($\boldsymbol{s}$)                               ▷ indirect sort
return $\bar{\boldsymbol{H}}, \boldsymbol{\omega}$
```

It is important to note that the two functions used in this pseudocode to compute the homography and similarity matrices stand for arbitrary methods that produce the required transformations.

Our score function (7) is just a proxy for the reprojection error computed over the whole image. Since we utilize only a small subset of points from the entire image, which may be subjected to noise, the assumption that the "best" homography is the one our method ranks as first may not hold in every case. In very few cases, the marker that achieves the lowest score function value does indeed reconstruct the remaining markers the best but not the overall image. However, our experiments show that our method consistently preserves its performance under various conditions.

4. Experiments

We evaluated the proposed homography ranking algorithm in various conditions. We tested cases involving various similarity transformations applied to original markers as well as noisy point correspondence, e.g., errors in marker detection since these are the expected problems in real-world scenarios.

4.1. Implementation Details

Our proposed algorithm can extend any homography estimation method that exploits point correspondences. For demonstration, we adopted time-tested implementations from the OpenCV 4.4.0 library [19]. Each homography was estimated by the `findHomography()` function, which employs the DLT [45] algorithm for $k = 4$ and the RANSAC [18] algorithm for $k > 4$, where k is the size of the point correspondences set. Each optimal similarity transformation between two 2D point sets was estimated by the `estimateAffinePartial2D()`, which also utilizes RANSAC for robustness. We always used default parameters.

4.2. Dataset Creation

We created a synthetic dataset to simulate the presence of markers in the scene subjected to perspective distortion. Our experiments were based on a pixel-wise comparison of the reprojection error. The synthetic dataset covered multiple setups named the test scenarios. For each test scenario, we generated t different samples, which we refer to as test instances. We set $t = 1\,000$. Table 1 contains description of the generated test scenarios. To create test instances (within test scenarios), we employed the procedures described below (see Figure 4).

Table 1. Description of the test scenarios in our synthetic dataset with corresponding settings and results for the top-ranked homography. One row represents one test scenario. Four visually separated groups (from top to bottom) are related to experiments shown in Figures 6–9.

Shape	Markers	Transl.	Rotation	Scale	Noise	Top Relative Improvement			Top Absolute Improvement		
						Median	Mean	Stdev	Median	Mean	Stdev
square	6	no	no	no	no	62.80%	59.63%	19.64%	0.00029	0.00030	0.00014
square	6	yes	no	no	no	62.65%	59.00%	19.72%	0.00028	0.00029	0.00013
square	6	no	yes	no	no	66.42%	63.17%	19.11%	0.00041	0.00043	0.00020
square	6	no	no	yes	no	63.38%	58.51%	23.97%	0.00024	0.00025	0.00015
square	6	yes	yes	yes	no	67.82%	63.66%	20.30%	0.00035	0.00037	0.00019
square	6	yes	yes	yes	yes	64.11%	59.26%	22.12%	22.07813	24.31773	15.00850
5-poly	6	yes	yes	yes	yes	74.67%	71.19%	21.98%	69.55532	336.26534	685.74274
7-poly	6	yes	yes	yes	yes	71.02%	65.63%	22.99%	46.79390	135.65737	395.75257
9-poly	6	yes	yes	yes	yes	68.97%	65.57%	21.98%	44.97627	115.12189	309.27201
square	3	yes	yes	yes	yes	46.91%	41.36%	31.58%	14.77504	18.11548	20.67457
square	5	yes	yes	yes	yes	59.03%	53.91%	24.56%	19.76285	22.53333	16.00804
square	7	yes	yes	yes	yes	66.19%	62.41%	19.98%	23.87681	27.13637	32.28533
square	9	yes	yes	yes	yes	69.86%	66.09%	18.18%	25.66452	26.68378	11.69754

We organized the creation of our dataset to allow for complete reproducibility of the reported results. Thanks to the synthetic nature of our data, fixing the seed for the used pseudo-random generator was sufficient. The source code for running the experiments is freely available (see the online Supplementary Materials at the end).

4.2.1. Image Initialization

Each test instance was initialized as a blank 1024 × 768 image. This image served for m randomly generated copies of the same shape (marker) placed in a 3×3 grid, where $0 < m \leq 9$. We used a uniform border with 20% size of the corresponding side to prevent the generated shapes from reaching outside of the image. We experimented with a different number of markers. From the set of 3×3 possible anchors, we chose m randomly, onto which we placed the generated markers. We also studied the effect of 3, 5, 7, and 9 out of 9 possible markers, given that all of the similarity transformations and noise were applied. Regarding marker shapes, we tested squares or convex, equilateral polygons with a tight bounding box of size 100 × 100 pixels (covering approximately 1.3% of the image). However, other similar shapes could be used, too. Their centroids were evenly distributed over the image, whilst the grid cells served as anchors. We adopted random generators from a uniform probability distribution. These settings represented the default configuration. Subsequently, we applied further transformations to the generated markers and the image.

4.2.2. Similarity Transformation

We showed the effect of similarity transformations before applying the perspective transformation. The translation and rotation demonstrate that markers could be positioned arbitrarily in a real environment provided that they shared the same planar surface. The change in scale showed that markers could be of different sizes.

To simulate a similarity transformation, we applied random rotation from the interval $[0, 360)$ degrees with the origin in the marker center. Then, we generated a random coordinate shift from interval $[-20, 20]$ pixels for translation in the x and y directions. However, an identical translation had to be applied to the entire marker to prevent distortion. Then, uniform scaling was performed with the origin in the marker center with a scale factor randomly generated from the interval $[0.8, 1.5]$. Due to this range, a ratio of the marker to image area ranged from 1.0% to 1.9%.

4.2.3. Perspective Distortion

We simulated a 3D rotation of an image around its center to represent a change in perspective on the plane that contained several markers. We rotated the image around its center in the x, y, and z axis by a random angle from interval $[-20, 20]$ degrees to achieve a change in perspective. The original keypoints were transformed along with the entire image, producing the warped keypoints.

4.2.4. Noisy Point Correspondence

To simulate a noisy point correspondence, we applied a random noise (translation) to each x and y coordinate of the warped keypoints from the interval $[-2, 2]$ pixels. At this stage, each keypoint was modified in isolation to achieve the distortive effect. Thanks to the perspective deformation, the generated random shift represented different levels of noise depending on how much the image had been warped. This step imitated errors in the marker detection, leading to a noisy point correspondence.

4.3. Evaluation Methodology

4.3.1. Error Computation

We evaluated the accuracy of our method by measuring the reprojection error using the Euclidean distance between the original and the rectified pixel positions. To obtain an error over the entire image, we computed the error for each pixel. Let w and h be the width and height of the image, respectively. The 3D rotation of a point in the image around the image center that produces perspective distortion is represented by $\varphi(\cdot)$. Let $\mathbf{g}_{i,j}^T = [j, i, 1]$ be the original (ground-truth) pixel position at the ith row and jth column, and let $\mathbf{w}_{i,j} = \varphi(\mathbf{g}_{i,j})$ be the analogically defined warped pixel position, for $i = 1, \ldots, h, j = 1, \ldots, w$. We then compute the 2D reprojection error grid (a $h \times w$ matrix) for the given homography $\mathbf{H}$ as

$$\xi_{wh} = \begin{bmatrix} e(\mathbf{w}_{1,1}, \mathbf{g}_{1,1}) & \ldots & e(\mathbf{w}_{1,w}, \mathbf{g}_{1,w}) \\ \ldots & \ldots & \ldots \\ e(\mathbf{w}_{h,1}, \mathbf{g}_{h,1}) & \ldots & e(\mathbf{w}_{h,w}, \mathbf{g}_{h,w}) \end{bmatrix}, \tag{9}$$

where

$$e(\mathbf{w}, \mathbf{g}) = \|\mathbf{H}\mathbf{w} - \mathbf{g}\|_2. \tag{10}$$

To express the reprojection error as a single number for the whole image, we adopted an arithmetic mean of all the values in the error grid above, so

$$\xi_{\text{reproj}} = \frac{1}{wh} \sum_{i=1}^{h} \sum_{j=1}^{w} e(\mathbf{w}_{i,j}, \mathbf{g}_{i,j}). \tag{11}$$

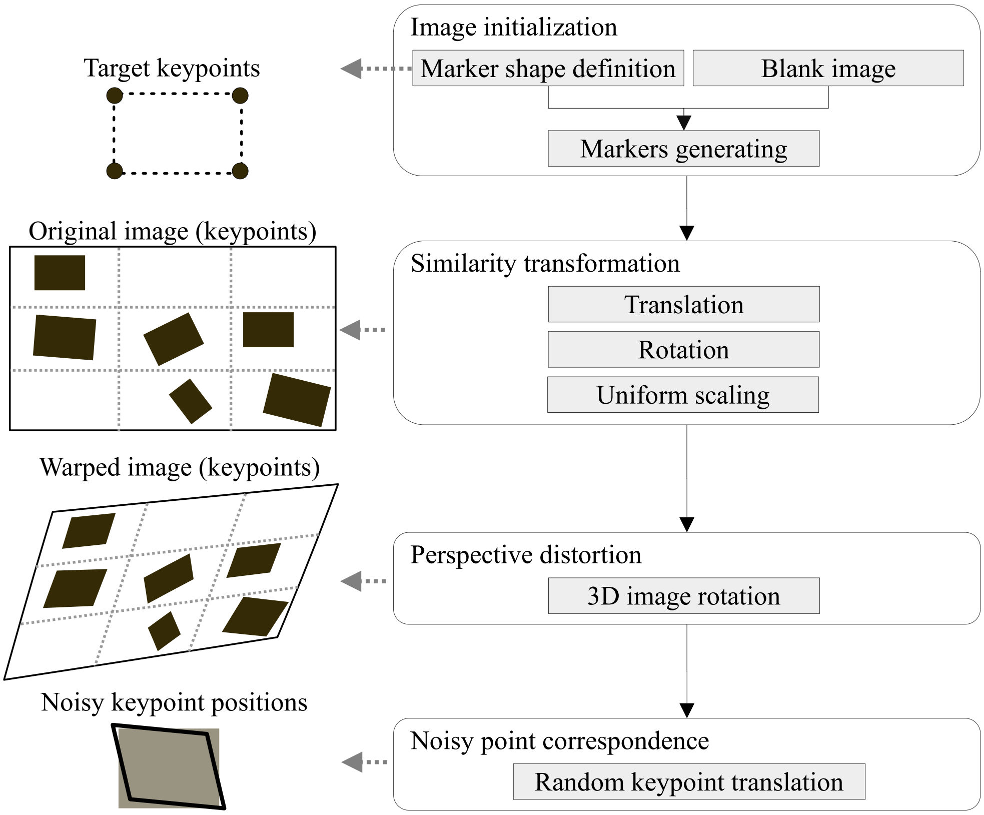

Figure 4. The description of how each one of the t test instances in a specific test scenario is created. The input is a blank $w \times h$ image over which m markers are initialized in a uniform grid, which produces the original marker keypoints. Depending on the test scenario, a particular subset of similarity transformations is applied to the entire image. Subsequently, warped keypoints are modified by random noise to simulate noisy point correspondence.

4.3.2. Evaluation Algorithm

On the input, we have m markers (Section 4.2) and thus an m-to-1 point correspondence. Each marker provides its unique homography. Our goal is to quantify the relative improvement in the reprojection error over the baseline when the kth ranked homography is used for rectification. Even though we are primarily concerned only with the single, top-performing homography, we evaluate the entire ranking to demonstrate stable behavior.

We evaluated our homography ranking in terms of reprojection error improvements against the existing approaches based on the isolated homography estimation represented by OpenCV [19] implementation. Since our method provides a ranking, we compare our performance against a random marker selection based on uniform probability distribution. We refer to this performance as the “baseline”, an unbiased marker selection. To obtain the aforementioned baseline, we evaluated the reprojection error (11) for each marker in isolation and computed the arithmetic mean of these values. When we executed our proposed algorithm, we obtained the full ordering of markers by their score value computed using the proposed criterion (7). We expected that, if the first marker is used to rectify the image, then the reprojection error is minimal (and lower than the baseline error). If any subsequent marker in the given order is used instead, the reprojection error increases.

We computed the relative improvement in % for each kth homography according to the baseline performance. Each test scenario was evaluated separately. For each test instance, we obtained a k-dimensional vector, where its elements represented percentual improvement at each kth position. We represented our data as a $t \times k$ matrix, where t was the number of test instances. We treated each column separately to compute the statistics. Our evaluation algorithm is described in Algorithm 2. For simplicity, we show an evaluation of a single instance.

Algorithm 2 Evaluation algorithm.

```
$\bar{\boldsymbol{H}}, \boldsymbol{\omega} \leftarrow$ RANKHOMOGRAPHIES( )          ▷ Algorithm 1
$e_b \leftarrow 0$                                          ▷ baseline
$\boldsymbol{e} \leftarrow$ array$[m]$                                   ▷ reprojection errors
$\boldsymbol{p} \leftarrow$ array$[m]$                                   ▷ relative improvements
for $i \leftarrow 1, \ldots, m$ do
    $\boldsymbol{e}[i] \leftarrow \xi_{\text{reproj}}$                            ▷ Equation (11)
    $e_b \leftarrow e_b + \boldsymbol{e}[i]$
end for
$e_b \leftarrow e_b / m$                                      ▷ mean reprojection error
for $i \leftarrow 1, \ldots, m$ do
    $k \leftarrow \boldsymbol{\omega}[i]$                                  ▷ position of $i$-th best homography
    $\boldsymbol{p}[i] \leftarrow (e_b - \boldsymbol{e}[k]) / e_b$                    ▷ relative improvement
end for
return $\boldsymbol{p}$
```

4.4. Results

Figure 5 shows how the reprojection error varies with respect to the marker position. We can see that the marker position can be deduced by looking at the heatmap representing the pixel-wise reprojection error over the image. The transformation achieves the best accuracy in the marker neighborhood and steadily decreases for more distant pixels. However, not all markers are subjected to the same pattern of error variation. This observation was the core motivation for our solution. We aim to choose the marker that minimizes the pixel-wise reprojection error within the region of the image that is as broad as possible. That is why we evaluate our method by computing the reprojection error over each pixel, not just the keypoints.

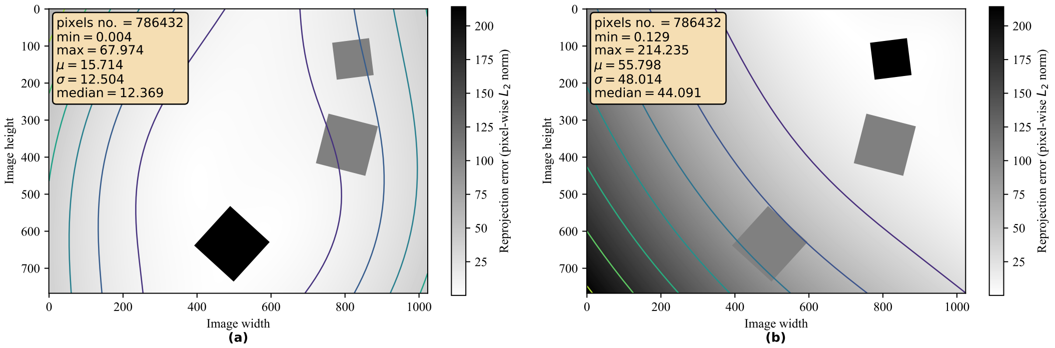

Figure 5. Distribution of pixel-wise reprojection error. The heat map together with corresponding contours demonstrate the varying distance between the ground truth and rectified pixel position after removing the perspective distortion. The bold square represents the reference marker. We show the result of (**a**) the "best" marker and (**b**) the "worst" marker. This test scenario includes all similarity transformations as well as noise in point correspondence.

All tested scenarios depict similar trends, as shown on the plots in Figures 6–9. The box plots extend from the lower to upper quartile values, with the thin and thick lines representing the median and mean, respectively. The plots discussed further show relative improvements over the baseline OpenCV [19] method. We evaluated relative improvements for the sake of interpretability. For better comprehension, we present Table 1. It contains individual test scenarios and their corresponding top performances in percents. Conversely, the reprojection error in absolute terms is difficult to interpret without additional context. Nevertheless, to highlight the differences in reprojection errors, we also provide absolute values in Table 1. The presence of noise shifted the errors by multiple magnitudes but still preserved the pattern of distribution.

4.4.1. Influence of Similarity Transformations

In this test scenario, we tested each allowed similarity transformation in isolation, i.e., translation, rotation, and uniform scaling. Figure 6 demonstrates that the relative improvement was circa equal in all situations. Moreover, we show that the proposed method is practically invariant to similarity transformations allowing the markers to be in arbitrary positions in a plane. When all similarity transformations were utilized, our method performed even better, showing its stability and robustness.

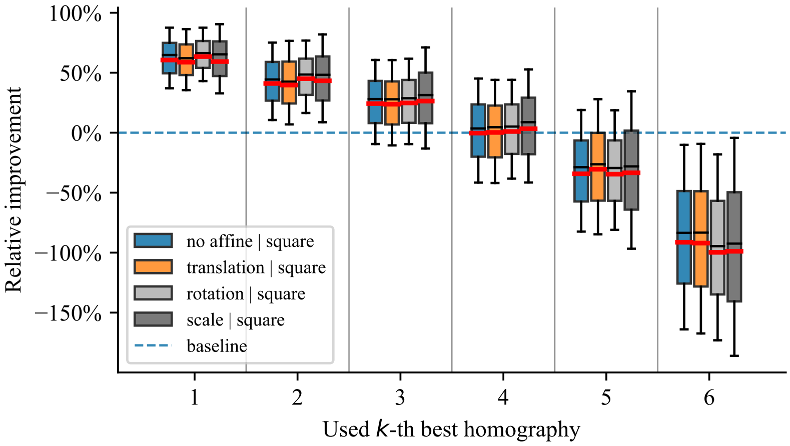

Figure 6. Influence of similarity transformation on the reprojection error.

4.4.2. Influence of Noise

In Figure 7, we can see the effect of a noisy point correspondence that simulated an inaccurate keypoint detection. The ranking method preserved the trend of the relative improvement in the presence of noise. The absolute reprojection error demonstrated that, unless noise was present, the errors varied on sub-pixel levels, so they were practically zero.

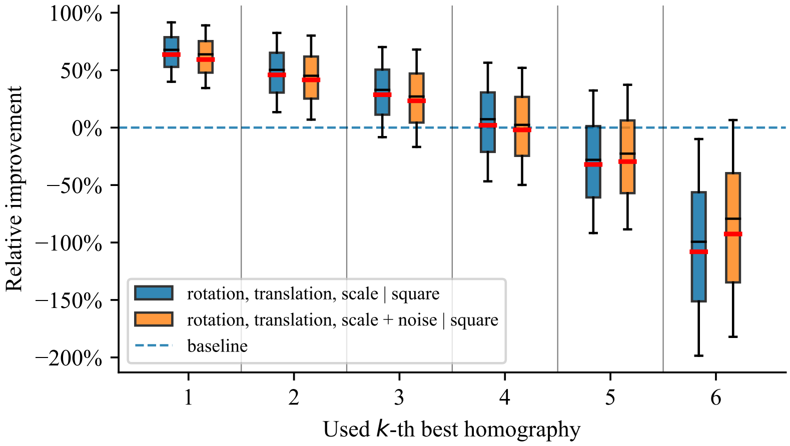

Figure 7. Influence of noise applied to the warped keypoints representing a noisy point correspondence.

4.4.3. Influence of Variable Shapes

We expected that the relative improvement of our method should be invariant to variable shapes as long as they were similar. Figure 8 demonstrates that, with an increasing number of keypoints, our method consistently preserved its capabilities. Introducing more complicated shapes than just rectangles did not exacerbate the outcome of the algorithm.

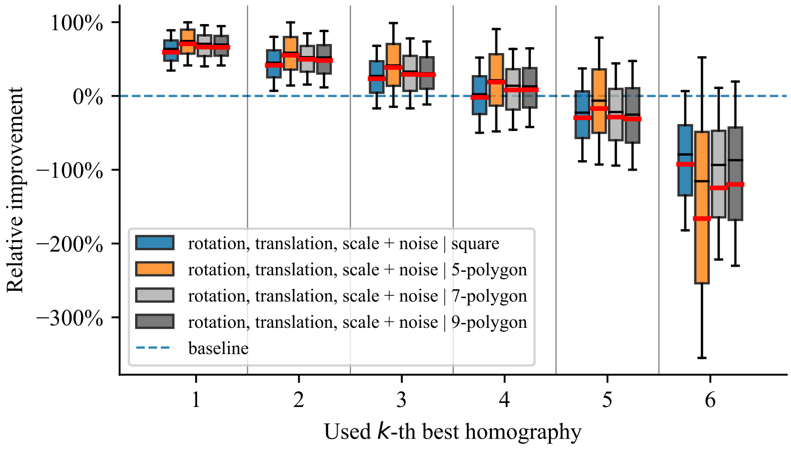

Figure 8. Results for different marker shapes.

4.4.4. Influence of Number of Markers

We tested a variable number of markers to demonstrate that our method preserved its improvement. Figure 9 shows that, the greater the set of markers, the better the relative improvement of our method. Even when we used just three markers, the proposed method achieved a 46.91% median relative improvement. While it is beneficial to use a larger number of markers, we believe that the improvement we can obtain from an increasing number of markers has a logarithmic trend. On the extreme side, if we used only one marker, there would be no improvement since there would be only one homography to choose from.

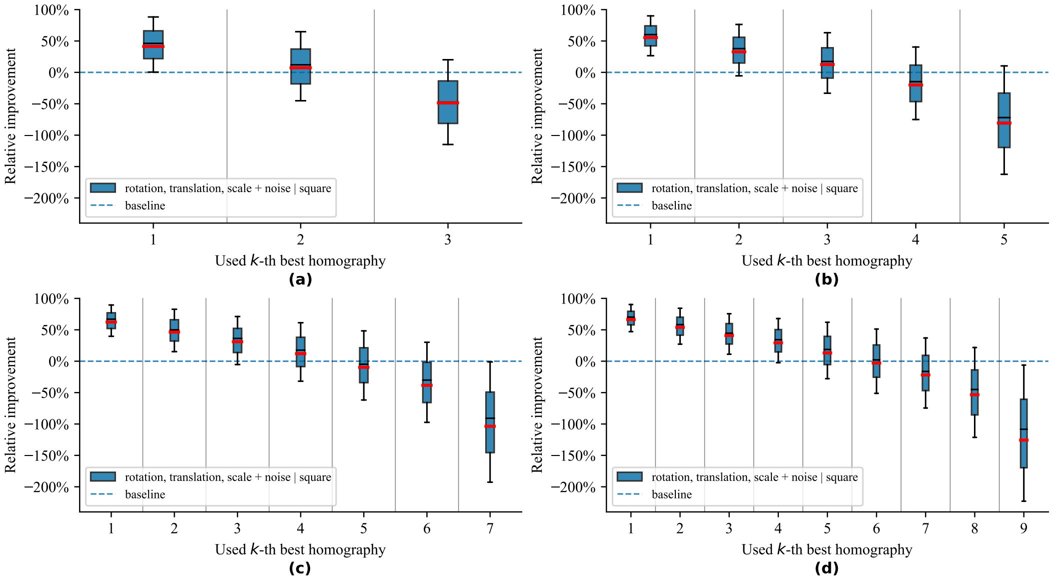

Figure 9. Influence of a different number of markers on reprojection error. We experimented with (**a**) three, (**b**) five, (**c**) seven, and (**d**) nine markers.

5. Conclusions

In this paper, we proposed a method that builds on top of existing approaches for homography estimation that utilize point correspondences. Our method systematically ranks a set of homography matrices according to our proposed score function. Each homography in this set belongs to a specific marker. These markers are objects of known shape either naturally occurring or purposely placed in the scene.

This method is based on three assumptions. The first is that the markers are geometrically similar, i.e., they differ only in translation, rotation, and uniform scale in the real world. The second is that the shape of at least one of them is known. The third is that these markers are placed on the same planar surface in the scene. Our approach shows a way to relate all of the markers to each other in a single score function without knowing their relative positions in the scene. Our method only handles transformation from a distorted to the undistorted view of the target plane. Thus, it serves the removal of perspective distortion.

We exploited the properties of homography and similarity transformations and expressed them in a single score function. This function stands at the core of our contribution. Its value is used as a proxy to rank homographies according to their reprojection error over the entire image using only markers' keypoints. The usual use case would be to select the homography with the lowest score, i.e., the highest-ranked matrix, to perform the image rectification.

We demonstrated that the proposed solution is robust in the presence of noise in the point correspondences. These correspondences can be either algorithmically found using feature-matching algorithms (e.g., SIFT [46] and SURF [47]) or annotated manually. However, even human annotations are often inaccurate. We also showed the robustness of our method to a varying number of markers and a change in shape.

All of our test scenarios demonstrated the following trend. On average, the homography with the highest score improved the relative performance to the baseline performance the most (both median and mean above 60%). The lowest-ranked homography often led to a lot worse performance (median and mean around −90%). These values varied slightly across different setups. The shape and number of markers had the greatest influence. All of the improvements in between steadily decreased and reached 0% improvement at around $^2/_3\ m$, where m is the number of markers. A general claim is that the first half of ranked homographies yields a better reprojection compared with the baseline on average. The baseline performance was given by an average OpenCV [19] reprojection error under the assumption of no prior preference of specific markers, hence the random marker selection.

Our algorithm is invariant to the underlying homography estimation method. It can thus serve as an extension to approaches that handle point correspondences, either as part of run time or a post-processing stage. Moreover, it is computationally very efficient, as it scales well with a quadratic complexity $\Theta\left(m^2\right)$.

Supplementary Materials: The following are available online at https://github.com/mondrasovic/homography_ranking, accessed on 26 August 2021.

Author Contributions: Conceptualization, M.O. and P.T.; methodology, M.O. and P.T; software, M.O.; validation, M.O. and P.T.; formal analysis, M.O. and P.T.; investigation, M.O.; resources, M.O.; data curation, M.O.; writing—original draft preparation, M.O.; writing—review and editing, P.T.; visualization, M.O. and P.T.; supervision, P.T.; project administration, M.O. and P.T.; funding acquisition, P.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by VEGA grant number 1/0689/19 and Grant System of University of Žilina number 1/2020 (8074).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Our study involved only synthetic data generated at runtime during the execution of our experiments. We provided the link to the source code of our entire pipeline in Supplementary Materials above.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A

Appendix A.1. Method Details

In this part, we provide more mathematical details concerning our score function defined in Equation (7). It is based on the idea that the knowledge of the homography allows us to remove the perspective distortion and to then apply any similarity transformation, i.e., translation, rotation, and uniform scaling [44]. We can transform the perspectively distorted view of the plane into a rectified form where parallelism and the ratio of lengths and angles are preserved.

A homography is a perspective projection between different views of a planar surface; hence, a 3D-to-2D transformation is reduced to a 2D-to-2D transformation. However, if the planar surface is frontal, i.e., it is perpendicular to the optical axis of the camera, then homography reduces to similarity transformation. Consequently, if we rectify the image, then the only allowed transformations are similarity transformations that we exploit in the computation of our score function. Moreover, this also explains why the shear has to be omitted.

Let $\mathbf{p}^T = [x, y, z, 1]$ be an arbitrary point that lies on a frontal plane given in world coordinates. Suppose that the optical axis of the camera is represented by the Z-axis. Then, $\mathbf{p}^T = [x, y, 0, 1]$. A projection of the point $\mathbf{p}$ onto a point $[\tilde{u}, \tilde{v}, 1]^T$ in the image plane using the pinhole camera model is given by

$$\begin{bmatrix} \tilde{u} \\ \tilde{v} \\ 1 \end{bmatrix} \approx \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 0 \\ 1 \end{bmatrix}, \tag{A1}$$

which simplifies to

$$\begin{bmatrix} \tilde{u} \\ \tilde{v} \\ 1 \end{bmatrix} \approx \begin{bmatrix} fr_{11} & fr_{12} & ft_x \\ fr_{21} & fr_{22} & ft_y \\ r_{31} & r_{32} & t_z \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, \tag{A2}$$

where f is the scale (focal length); r_{ij} for $i, j = 1, \ldots, 3$ specifies the rotation; and t_x, t_y, and t_z denote the translation. The 3D rotation is reduced to a 2D rotation:

$$\begin{bmatrix} \tilde{u} \\ \tilde{v} \\ 1 \end{bmatrix} \approx \begin{bmatrix} f\cos(\theta) & -f\sin(\theta) & ft_x \\ f\sin(\theta) & f\cos(\theta) & ft_y \\ 0 & 0 & t_z \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, \tag{A3}$$

yielding a similarity transformation. Therefore, our score function needs to encompass the same similarity transformations that could affect the objects on the planar surface in the real world.

Consider the following hierarchy of transformations: similarity, affine, and projective. A projective transformation can be decomposed into a chain of transformations, where each matrix is given by a transformation that is higher in the hierarchy than the previous one [3,29]. Specifically, a homography $\mathbf{H}$ may be decomposed into similarity, affinity, and projectivity as follows:

$$\mathbf{H} = \mathbf{H}_S \mathbf{H}_A \mathbf{H}_P = \begin{bmatrix} s\mathbf{R} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{bmatrix} \mathbf{K} & \mathbf{0} \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{v}^T & v \end{bmatrix}, \tag{A4}$$

such that $v \neq 0$ [3].

Our score function exploits the transformation of the plane into a frontal plane that is related to the ground-truth plane by similarity. It uses a homography followed by a similarity transformation to quantify the reprojection error for keypoint rectification. Besides the empirical evidence, the transformation using a homography followed by a similarity transformation has a simple theoretical justification.

The similarity matrix is a special case of the homography matrix. According to Equation (A4), it may be decomposed as

$$\mathbf{S} = \mathbf{H}_S^{(S)} \mathbf{H}_A^{(S)} \mathbf{H}_P^{(S)} = \mathbf{H}_S^{(S)} \cdot \mathbf{I} \cdot \mathbf{I}. \tag{A5}$$

A composition of linear transformations gives us

$$\mathbf{H}' = \mathbf{S}\mathbf{H} = \mathbf{H}_S^{(S)}\mathbf{H} = \left(\mathbf{H}_S^{(S)}\mathbf{H}_S\right)\mathbf{H}_A\mathbf{H}_P = \mathbf{H}'_S\mathbf{H}_A\mathbf{H}_P, \tag{A6}$$

which is a homography transformation, too.

Appendix A.2. Joint Optimization

Our score function defined in (7) evaluates the reprojection error of multiple markers when a similarity transformation is applied after rectification. Given the homography ranking approach introduced in Algorithm 1, we observe that the required transformation matrices can be either retrieved from the input or estimated in isolation at run time. We attempted to perform a joint optimization where the homography matrix **H**, together with the set of similarity matrices $\mathcal{S}$, were estimated simultaneously. Thus, all DoFs were treated as decision variables.

Our optimization objective function was the unmodified score function from Equation (7). We used the L-BFGS algorithm implemented in PyTorch [48], which is an iterative method for solving unconstrained nonlinear optimization problems [49] with good performance for non-smooth optimizations [26,50]. It utilizes the first-order gradients and the estimated inverse Hessian matrix. We obtained the gradients using automatic differentiation [48]. The initial estimates for the homography and the similarity matrices were obtained using the OpenCV [19] implementations (see Section 4.1).

The optimization was unstable because the algorithm was allowed to update all of the unknowns simultaneously, which led to exploding gradients. We thus split the optimization into two parts: isolated refining of the homography while having the coefficients for similarity transformations frozen. Then, the roles changed, and similarity transformations were refined while keeping the homography untouched. This sometimes resulted in degenerate homography matrices, i.e., their determinant was negative or very close to zero [28]. We then expanded the objective by adding a penalty term for the determinant value. It stabilized the optimization, improved convergence, and produced usable homographies. However, this joint optimization brought either no or just a minor improvement (approximately 3%) in the reprojection error. Moreover, the computational overhead was substantial. We believe that, since the proposed score function incorporates all of the markers, it brings the greatest improvement by itself. Further refinement of the involved transformations will probably have diminishing returns.

References

1. Geetha Kiran, A.; Murali, S. Automatic rectification of perspective distortion from a single image using plane homography. *J. Comput. Sci. Appl.* **2013**, *3*, 47–58.
2. Bousaid, A.; Theodoridis, T.; Nefti-Meziani, S.; Davis, S. Perspective distortion modeling for image measurements. *IEEE Access* **2020**, *8*, 15322–15331. [CrossRef]
3. Hartley, R.; Zisserman, A. *Multiple View Geometry in Computer Vision*, 2nd ed.; Cambridge University Press: Cambridge, UK, 2003.
4. Hartley, R.I. In defense of the eight-point algorithm. *IEEE Trans. Pattern Anal. Mach. Intell.* **1997**, *19*, 580–593. [CrossRef]
5. Lu, S.; Chen, B.M.; Ko, C.C. Perspective rectification of document images using fuzzy set and morphological operations. *Image Vis. Comput.* **2005**, *23*, 541–553. [CrossRef]
6. Miao, L.; Peng, S. Perspective rectification of document images based on morphology. In Proceedings of the 2006 International Conference on Computational Intelligence and Security, Guangzhou, China, 3–6 November 2006; Volume 2, pp. 1805–1808. [CrossRef]
7. Adel, E.; Elmogy, M.; Elbakry, H. Image stitching based on feature extraction techniques: A survey. *Int. J. Comput. Appl.* **2014**, *99*, 1–8. [CrossRef]
8. Gao, J.; Kim, S.J.; Brown, M.S. Constructing image panoramas using dual-homography warping. In Proceedings of the CVPR 2011, Colorado Springs, CO, USA, 20–25 June 2011; pp. 49–56. [CrossRef]
9. Liu, W.X.; Chin, T. Smooth Globally Warp Locally: Video Stabilization Using Homography Fields. In Proceedings of the 2015 International Conference on Digital Image Computing: Techniques and Applications (DICTA), Adelaide, SA, Australia, 23–25 November 2015; pp. 1–8. [CrossRef]
10. Zhang, Z. A flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell.* **2000**, *22*, 1330–1334. [CrossRef]

11. Mariyanayagam, D.; Gurdjos, P.; Chambon, S.; Brunet, F.; Charvillat, V. Pose estimation of a single circle using default intrinsic calibration. In *Asian Conference on Computer Vision*; Springer: Cham, Switzerland, 2018.
12. Arróspide, J.; Salgado, L.; Nieto, M.; Mohedano, R. Homography-based ground plane detection using a single on-board camera. *IET Intell. Transp. Syst.* **2010**, *4*, 149–160. [CrossRef]
13. Luo, L.B.; Koh, I.S.; Min, K.Y.; Wang, J.; Chong, J.W. Low-cost implementation of bird's-eye view system for camera-on-vehicle. In Proceedings of the 2010 Digest of Technical Papers International Conference on Consumer Electronics (ICCE), Las Vegas, NV, USA, 9–13 January 2010; pp. 311–312. [CrossRef]
14. Wang, Y.; Yu, M.; Jiang, G.; Pan, Z.; Lin, J. Image Registration Algorithm Based on Convolutional Neural Network and Local Homography Transformation. *Appl. Sci.* **2020**, *10*, 732. [CrossRef]
15. Zhang, Y.; Zhou, L.; Liu, H.; Shang, Y. A Flexible Online Camera Calibration Using Line Segments. *J. Sens.* **2016**, *2016*, 2802343. [CrossRef]
16. Osuna-Enciso, V.; Cuevas, E.; Oliva, D.; Zúñiga, V.; Pérez-Cisneros, M.; Zaldívar, D. A Multiobjective Approach to Homography Estimation. *Comput. Intell. Neurosci.* **2015**, *2016*, 3629174. [CrossRef]
17. Mou, W.; Wang, H.; Seet, G.; Zhou, L. Robust homography estimation based on non-linear least squares optimization. In Proceedings of the 2013 IEEE International Conference on Robotics and Biomimetics (ROBIO), Shenzhen, China, 12–14 December 2013; pp. 372–377. [CrossRef]
18. Fischler, M.A.; Bolles, R.C. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Commun. ACM* **1981**, *24*, 381–395. [CrossRef]
19. Bradski, G.; Kaehler, A. *Learning OpenCV: Computer Vision with the OpenCV Library*; O'Reilly Media, Inc.: Newton, MA, USA, 2008.
20. Agarwal, A.; Jawahar, C.; Narayanan, P. *A Survey of Planar Homography Estimation Techniques*; Technical Report IIIT/TR/2005/12; Centre for Visual Information Technology: Telangana, India, 2005.
21. Benligiray, B.; Topal, C.; Akinlar, C. STag: A stable fiducial marker system. *Image Vis. Comput.* **2019**, *89*, 158–169. [CrossRef]
22. Zhu, H.; Wen, X.; Zhang, F.; Wang, X.; Wang, G. Homography estimation based on order-preserving constraint and similarity measurement. *IEEE Access* **2018**, *6*, 28680–28690. [CrossRef]
23. Jawahar, C.; Jain, P. Homography estimation from planar contours. In Proceedings of the Third International Symposium on 3D Data Processing, Visualization, and Transmission (3DPVT'06), Chapel Hill, NC, USA, 14–16 June 2006; pp. 877–884. [CrossRef]
24. Chen, Y.; Sun, J.; Wang, G. Minimizing Geometric Distance by Iterative Linear Optimization. In Proceedings of the 2010 20th International Conference on Pattern Recognition, Istanbul, Turkey, 23–26 August 2010; pp. 1–4. [CrossRef]
25. Chum, O.; Pajdla, T.; Sturm, P. The geometric error for homographies. *Comput. Vis. Image Underst.* **2005**, *97*, 86–102. [CrossRef]
26. Li, T.M.; Gharbi, M.; Adams, A.; Durand, F.; Ragan-Kelley, J. Differentiable programming for image processing and deep learning in Halide. *ACM Trans. Graph.* **2018**, *37*, 1–13. [CrossRef]
27. Song, W.H.; Jung, H.G.; Gwak, I.Y.; Lee, S.W. Oblique aerial image matching based on iterative simulation and homography evaluation. *Pattern Recognit.* **2019**, *87*, 317–331. [CrossRef]
28. Vincent, E.; Laganiére, R. Detecting planar homographies in an image pair. In Proceedings of the ISPA 2001, the 2nd International Symposium on Image and Signal Processing and Analysis and 23rd International Conference on Information Technology Interfaces, Pula, Croatia, 19–21 July 2001; pp. 182–187. [CrossRef]
29. Bose, B.; Grimson, E. Ground plane rectification by tracking moving objects. In Proceedings of the e Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance, Lausanne, Switzerland, 28 July 2003; Volume 7.
30. Eriksson, A.; Van Den Hengel, A. Optimization on the manifold of multiple homographies. In Proceedings of the 2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops, Kyoto, Japan, 27 September–4 October 2009; pp. 242–249. doi:10.1109/ICCVW.2009.5457692. [CrossRef]
31. Ruiz, A.; López-de Teruel, P.E.; Fernández, L. Practical Planar Metric Rectification. In Proceedings of the BMVC, Edinburgh, UK, 4–7 September 2006; pp. 579–588.
32. Pirchheim, C.; Reitmayr, G. Homography-based planar mapping and tracking for mobile phones. In Proceedings of the 2011 10th IEEE International Symposium on Mixed and Augmented Reality, Basel, Switzerland, 26–29 October 2011; pp. 27–36. [CrossRef]
33. Chojnacki, W.; Szpak, Z.L.; Brooks, M.J.; Van Den Hengel, A. Multiple homography estimation with full consistency constraints. In Proceedings of the 2010 International Conference on Digital Image Computing: Techniques and Applications, Sydney, NSW, Australia, 1–3 December 2010; pp. 480–485. [CrossRef]
34. Park, K.w.; Shim, Y.J.; Lee, M.j.; Ahn, H. Multi-Frame Based Homography Estimation for Video Stitching in Static Camera Environments. *Sensors* **2020**, *20*, 92. [CrossRef]
35. Cui, Z.; Jiang, K.; Wang, T. Unsupervised Moving Object Segmentation from Stationary or Moving Camera Based on Multi-frame Homography Constraints. *Sensors* **2019**, *19*, 4344. [CrossRef] [PubMed]
36. Fraundorfer, F.; Schindler, K.; Bischof, H. Piecewise planar scene reconstruction from sparse correspondences. *Image Vis. Comput.* **2006**, *24*, 395–406. [CrossRef]
37. DeTone, D.; Malisiewicz, T.; Rabinovich, A. Deep Image Homography Estimation. *arXiv* **2016**, arXiv:1606.03798.
38. Zhang, J.; Wang, C.; Liu, S.; Jia, L.; Ye, N.; Wang, J.; Zhou, J.; Sun, J. Content-Aware Unsupervised Deep Homography Estimation. In *Proceedings of the Computer Vision—ECCV 2020, Glasgow, UK, 23–28 August 2020*; Vedaldi, A., Bischof, H., Brox, T., Frahm, J.M., Eds.; Springer International Publishing: Cham, Switzerland, 2020; pp. 653–669.

39. Le, H.; Liu, F.; Zhang, S.; Agarwala, A. Deep Homography Estimation for Dynamic Scenes. *arXiv* **2020**, arXiv:2004.02132.
40. Zhao, Q.; Ma, Y.; Zhu, C.; Yao, C.; Feng, B.; Dai, F. Image stitching via deep homography estimation. *Neurocomputing* **2021**, *450*, 219–229. [CrossRef]
41. Tao, Y.; Ling, Z. Deep Features Homography Transformation Fusion Network—A Universal Foreground Segmentation Algorithm for PTZ Cameras and a Comparative Study. *Sensors* **2020**, *20*, 3420. [CrossRef]
42. Zhou, Q.; Li, X. STN-Homography: Direct Estimation of Homography Parameters for Image Pairs. *Appl. Sci.* **2019**, *9*, 5187. [CrossRef]
43. Barath, D.; Hajder, L. Novel Ways to Estimate Homography from Local Affine Transformations. In *Proceedings of the 11th Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2016), Rome, Italy, 27–29 February 2016*; Magnenat-Thalmann, N., Richard, P., Linsen, L., Telea, A.C., Battiato, S., Imai, F.H., Braz, J., Eds.; Springer: Berlin/Heidelberg, Germany, 2016; Volume 3, pp. 434–445. [CrossRef]
44. Beck, G. Planar Homography Estimation from Traffic Streams via Energy Functional Minimization. Ph.D. Thesis, Johns Hopkins University, Baltimore, MD, USA, 2016.
45. Abdel-Aziz, Y.; Karara, H.; Hauck, M. Direct Linear Transformation from Comparator Coordinates into Object Space Coordinates in Close-Range Photogrammetry. *Photogramm. Eng. Remote Sens.* **2015**, *81*, 103–107. [CrossRef]
46. Lowe, D.G. Object recognition from local scale-invariant features. In Proceedings of the Seventh IEEE International Conference on Computer Vision, Corfu, Greece, 20–27 September 1999; Volume 2, pp. 1150–1157. [CrossRef]
47. Bay, H.; Ess, A.; Tuytelaars, T.; Van Gool, L. Speeded-up robust features (SURF). *Comput. Vis. Image Underst.* **2008**, *110*, 346–359. [CrossRef]
48. Paszke, A.; Gross, S.; Chintala, S.; Chanan, G.; Yang, E.; DeVito, Z.; Lin, Z.; Desmaison, A.; Antiga, L.; Lerer, A. Automatic differentiation in PyTorch. In Proceedings of the NIPS 2017 Workshop Autodiff, NIPS-W, Long Beach, CA, USA, 9 December 2017.
49. Fletcher, R. Newton-Like Methods. In *Practical Methods of Optimization*; John Wiley and Sons, Ltd.: Hoboken, NJ, USA, 2000; Chapter 3, pp. 44–79. [CrossRef]
50. Nocedal, J.; Wright, S. *Numerical Optimization*; Springer Science & Business Media: Hoboken, NJ, USA, 2006.