sensors

MDPI

Article

Fast and Accurate Pose Estimation with Unknown Focal Length Using Line Correspondences

Kai Guo *, **Zhixiang Zhang, Zhongsen Zhang, Ye Tian and Honglin Chen**

Northwest Institute of Nuclear Technology, Xi'an 710024, China
* Correspondence: guokai@mail.nwpu.edu.cn; Tel.: +86-185-1023-7517

Abstract: Estimating camera pose is one of the key steps in computer vison, photogrammetry and SLAM (Simultaneous Localization and Mapping). It is mainly calculated based on the 2D–3D correspondences of features, including 2D–3D point and line correspondences. If a zoom lens is equipped, the focal length needs to be estimated simultaneously. In this paper, a new method of fast and accurate pose estimation with unknown focal length using two 2D–3D line correspondences and the camera position is proposed. Our core contribution is to convert the PnL (perspective-n-line) problem with 2D–3D line correspondences into an estimation problem with 3D–3D point correspondences. One 3D line and the camera position in the world frame can define a plane, the 2D line projection of the 3D line and the camera position in the camera frame can define another plane, and actually the two planes are the same plane, which is the key geometric characteristic in this paper's estimation of focal length and pose. We establish the transform between the normal vectors of the two planes with this characteristic, and this transform can be regarded as the camera projection of a 3D point. Then, the pose estimation using 2D–3D line correspondences is converted into pose estimation using 3D–3D point correspondences in intermediate frames, and, lastly, pose estimation can be finished quickly. In addition, using the property whereby the angle between two planes is invariant in both the camera frame and world frame, we can estimate the camera focal length quickly and accurately. Experimental results show that our proposed method has good performance in numerical stability, noise sensitivity and computational speed with synthetic data and real scenarios, and has strong robustness to camera position noise.

Keywords: pose estimation; line correspondences; unknown focal length; camera position; normal vector

check for updates

Citation: Guo, K.; Zhang, Z.; Zhang, Z.; Tian, Y.; Chen, H. Fast and Accurate Pose Estimation with Unknown Focal Length Using Line Correspondences. *Sensors* **2022**, *22*, 8253. https://doi.org/10.3390/s22218253

Academic Editor: Gwanggil Jeon

Received: 27 September 2022
Accepted: 26 October 2022
Published: 28 October 2022

1. Introduction

Camera pose estimation is an important step in computer vision, SLAM and photogrammetry [1–5], and the corresponding methods are mainly divided into two categories. The first category is based on deep learning using large training sets, which has become popular recently [6–9]. The second category is the traditional pose estimation method, which uses a small number of precise inputs for accurate estimation [10–15]. The former category is difficult to apply in high-precision measurements, and, in some real scenarios, such as weapon tests, it is difficult to obtain a large training set. The method proposed in this paper belongs to the latter, which uses a small number of precise inputs for pose estimation, and the most commonly used inputs here are 2D–3D point correspondences [16] and 2D–3D line correspondences [17]. Point and line features are common in real scenarios and are easy to obtain by humans; hence, there are many point-based pose estimation methods, called PnP (perspective-n-point) solvers [18–20], and line-based pose estimation methods, called PnL (perspective-n-line) solvers [21–24]. Moreover, there are some methods that estimate pose using both points and lines [25], but they are not applied widely. In addition, there are some specific methods using line correspondences that are parallel lines [26]. These parallel lines in the image plane would intersect at a point that is called the vanishing point. The specific methods seemingly use line correspondences, but, actually, we could also say that they use point correspondences (i.e., vanishing point). However, parallel lines do not always exist in real scenarios and they are also not applied widely, as with PnP or PnL solvers. Hence, here, we mainly introduce PnP and PnL solvers.

For PnP solvers, a 2D–3D point correspondence provides two constraints [27]. When all the intrinsic parameters are known and all the extrinsic parameters are unknown, at least three 2D–3D point correspondences are needed to estimate the camera pose, which is called a P3P solver [28], with up to four solutions, and an additional constraint (e.g., one more 2D–3D point correspondence) is needed to determine the unique solution. When there are four 2D–3D point correspondences, the camera pose and an intrinsic parameter (e.g., the focal length) can be simultaneously estimated, which is called P4Pf [29,30]. When five 2D–3D point correspondences are present, the camera pose and three intrinsic parameters (e.g., the focal length and radial distortion) can be simultaneously estimated, which is called P5Pfr [31]. All the above methods are nonlinear. When there are at least six 2D–3D point correspondences, the camera pose can be estimated linearly, which is called DLT (Direct Linear Transformation) [32,33]. In addition, there are some methods that use partial parameters of pose as prior knowledge, such as the known vertical direction [34–36] or camera position [37,38], and these methods can use fewer 2D–3D point correspondences, simplify the problem and increase the efficiency.

In some cases, no point features but line features are present. Similar to 2D–3D point correspondences, 2D–3D line correspondences are also widely used in pose estimation. The method proposed in this paper is an estimation method based on 2D–3D line correspondences. Similarly, a 2D–3D line correspondence can give two constraints. If all the intrinsic parameters are known and the six parameters of the pose are all unknown, at least three 2D–3D line correspondences are needed to estimate the pose, which is called P3L [39–41], and up to eight solutions can be obtained by solving nonlinear equations, and then another constraint is given to determine the unique solution. If some intrinsic parameters are unknown, more 2D–3D line correspondences are needed [42–44]. When there are four 2D–3D line correspondences, the focal length can be obtained, while the pose is estimated. When there are five 2D–3D line correspondences, the focal length and principal point (or radial distortion) can be obtained, while the pose is estimated. When there are at least six 2D–3D line correspondences, the camera pose can be linearly estimated [45]. As we can see, more parameters can be estimated using more 2D–3D line correspondences, and the computational complexity decreases (from nonlinear to linear estimation). However, sufficient 2D–3D line correspondences may not exist in the FOV (field of view), and a great deal of work is needed to measure the lines accurately. In this case, we need to finish the pose estimation with a smaller number of 2D–3D line correspondences, and this is the motivation for our work in this paper. If less than three 2D–3D line correspondences are required, some pose parameters need to be known. Some methods use IMU-based techniques to obtain the vertical direction [46]; that is, two angles are known, and then two 2D–3D line correspondences are used to estimate the pose. Similarly, based on this idea, we can measure the camera position in advance, which also reduces the number of 2D–3D line correspondences. With the development of technology, various tools are used to measure the camera position with high accuracy, such as total station and RTK (real-time kinematics). Therefore, in this paper, we propose a new method to simultaneously estimate the pose and focal length using two 2D–3D line correspondences and the camera position.

In this paper, two known 2D–3D line correspondences and the known camera position are used. In many cases, the camera position could be known as prior knowledge. In missile testing range and rocket launch applications, for example, the attitude measurement based on fixed cameras with a zoom lens is an important test. These cameras are fixed and the positions can be measured as prior knowledge by RTK or total station. The accuracy of the RTK and total station is high. In addition, with the growing prominence of the social security problem, VMCs (visual monitoring cameras) are used widely. In general, the position of the VMC is fixed and the lens orientation can be changed. In these cases, where the camera position is known, the presented method can be applied to estimate the focal length and pose using line correspondences. Each 3D line and camera position in the world frame can determine a plane, and then we can obtain two known planes in the world frame because two 3D lines are given. The angle between these two planes in the world frame can be calculated. Similarly, we can obtain two known planes in the camera frame because two 2D lines are given. Then, the angle with an unknown focal length, between these two planes in the camera frame, can be calculated. According to the principle of camera imaging, the two angles are equal and, consequently, an equation can be given with one parameter (i.e., focal length). Hence, the focal length can be estimated by using this geometric characteristic. In addition, the plane that contains the 3D line and camera position in the world frame, and the plane that contains the corresponding 2D line and camera position in the camera frame, are the same plane in space. Then, the unit normal vectors of the two planes can correspond through the rotation matrix, and, lastly, the unit normal vector correspondences are transformed into point correspondences under two intermediate frames with the same origin. Now, the pose estimation using line correspondences is transformed into pose estimation using point correspondences, and then we can efficiently estimate the camera pose using this characteristic. It can be seen that our proposed method uses only two 2D–3D line correspondences and the camera position to estimate the focal length and pose simultaneously, which improves the efficiency and expands the applicability of the proposed method while reducing the number of lines. Experimental results show that, compared with several existing pose estimation methods, our proposed method can achieve better performance in numerical stability, noise sensitivity and computational speed, both in synthetic data and real scenarios. In addition, our proposed method also has strong robustness to camera position noise.

The rest of this paper is as follows. Section 2 describes the materials and methods, and proposes the specific theoretical derivation of our proposed method. In Section 3, the numerical stability, noise sensitivity and computational speed of the proposed method and some other SOTA (state-of-the-art) methods are compared. Section 4 is the discussion; Section 5 is the conclusions.

2. Proposed Method

Two 3D lines L_1, L_2 and camera position O_c are known in the world frame S_{w1} ($O_w\_X_wY_wZ_w$). The projections of the two 3D lines are denoted as l_1, l_2 on the image plane and l_1, l_2 are known in the camera frame S_{c1} ($O_c\_X_cY_cZ_c$), as shown in Figure 1.

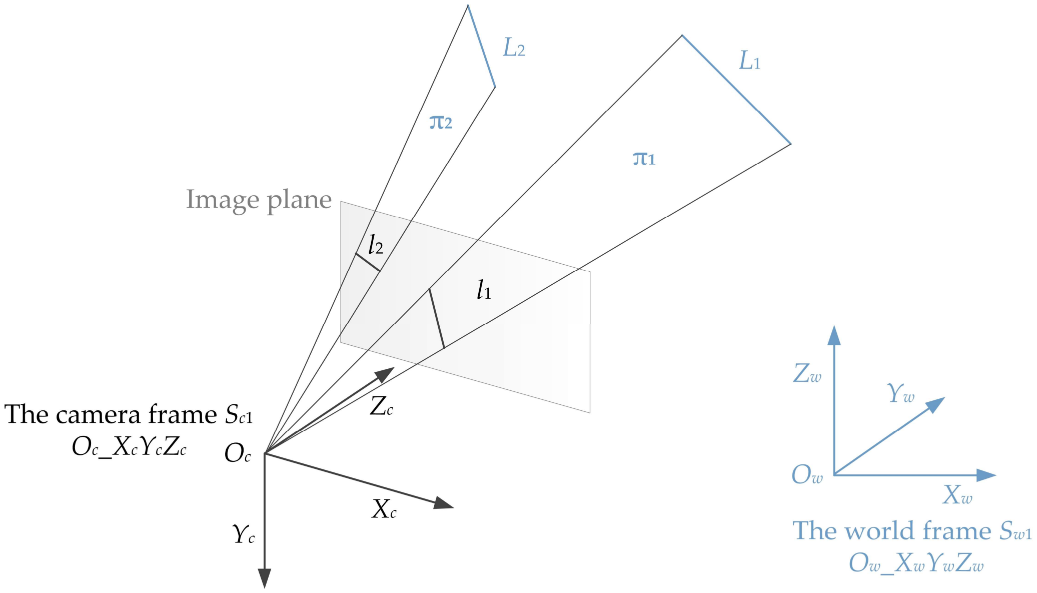

Figure 1. Two 3D lines and their projections in our proposed method.

Line L_i (i = 1, 2) and camera position O_c can determine a plane and then we can obtain two planes that are denoted as π_1, π_2 in this paper. Because the lines and camera position are known in the world frame, the planes π_1, π_2 are known in the world frame and then we can obtain two unit normal vectors of the two planes, denoted as $\boldsymbol{n_{w1}}$ and $\boldsymbol{n_{w2}}$, and they are known in the world frame S_{w1} ($O_w\_X_wY_wZ_w$). Similarly, by extracting two lines l_1, l_2 on the image plane, we can obtain two unit normal vectors $\boldsymbol{n_{c1}}$ and $\boldsymbol{n_{c2}}$ in the camera frame S_{c1} ($O_c\_X_cY_cZ_c$), but both contain an unknown variable, which is the focal length. The angle between the two normal vectors in the camera frame is equal to the angle in the world frame. Therefore, an equation can be established based on this relation among the angles. The equation contains only one unknown variable, namely the focal length, and then the focal length can be solved.

In addition, a vector in the world frame can be transformed into another vector in the camera frame with the camera rotation matrix. Hence, the unit normal vector $\boldsymbol{n_{wi}}$ in world frame S_{w1} ($O_w\_X_wY_wZ_w$) can be transformed into $\boldsymbol{n_{ci}}$ in camera frame S_{c1} ($O_c\_X_cY_cZ_c$) using $\boldsymbol{n_{c\mathrm{i}}} = \boldsymbol{Rn_{wi}}$; here, $\boldsymbol{R}$ is the camera rotation matrix, and contains all the pose information. According to this rotation relationship, we translate the origin of the world frame S_{w1} ($O_w\_X_wY_wZ_w$) to the camera position, and the new world frame S_{w2} ($O_{w2}\_X_{w2}Y_{w2}Z_{w2}$) is obtained. In this way, the projection relationship of unit normal vectors can be changed into the relationship between the point $\boldsymbol{n_{wi}}$ in the world frame S_{w2} ($O_{w2}\_X_{w2}Y_{w2}Z_{w2}$), and the point $\boldsymbol{n_{ci}}$ in the camera frame S_{c1} ($O_c\_X_cY_cZ_c$) through the rotation matrix R. R here is the rotation matrix between world frame S_{w2} ($O_{w2}\_X_{w2}Y_{w2}Z_{w2}$) and camera frame S_{c1} ($O_c\_X_cY_cZ_c$), and also the rotation matrix between world frame S_{w1} ($O_w\_X_wY_wZ_w$) and camera frame S_{c1} ($O_c\_X_cY_cZ_c$). After this, the problem with line correspondences has been converted into the problem with point correspondences, which means that the PnL problem in this paper has changed into the PnP problem. It is a useful transformation that makes the calculation process fast and efficient. The specific calculation process is as follows.

2.1. Focal Length Estimation

The expression of L_i is ($\boldsymbol{V_i}$, P_i) in this paper, and $\boldsymbol{V_i}$ is the unit direction vector of the line, while P_i is an arbitrary point on the line. Then, L_i in the world frame can be written as

$$\boldsymbol{L_i} = P_i + k\boldsymbol{V_i} \tag{1}$$

Here, k is an arbitrary scale factor, unknown. Then, in the world frame, the unit normal vector $\boldsymbol{n_{wi}}$ of the plane π_i passing through the line L_i and the camera position O_c is

$$\boldsymbol{n_{wi}} = \frac{\boldsymbol{V}_i \times \overrightarrow{P_iO_c}}{\|\boldsymbol{V_i} \times \overrightarrow{P_iO_c}\|} \tag{2}$$

The angle α between the unit normal vectors of the plane π_1 and π_2 can be obtained using

$$\alpha = \arccos\frac{\boldsymbol{n_{w1}^T} \times \boldsymbol{n_{w2}}}{\|\boldsymbol{n_{w1}}\|\cdot\|\boldsymbol{n_{w2}}\|} \tag{3}$$

Assume that the pixel coordinates of the two endpoints of the line l_i in the image plane of the corresponding L_i are given as $\begin{pmatrix} u_{2i-1} & v_{2i-1} \end{pmatrix}$, $\begin{pmatrix} u_{2i} & v_{2i} \end{pmatrix}$, which are known; the focal length of the camera is f, unknown; then, the normal vector of π_1 and π_2 in the camera frame is

$$\begin{aligned} \boldsymbol{n}_{c1} &= \begin{bmatrix} v_1 - v_2 & u_2 - u_1 & \frac{u_1v_2-u_2v_1}{f} \end{bmatrix} = \begin{bmatrix} a_1 & b_1 & \frac{c_1}{f} \end{bmatrix} \\ \boldsymbol{n}_{c2} &= \begin{bmatrix} v_3 - v_4 & u_4 - u_3 & \frac{u_3v_4-u_4v_3}{f} \end{bmatrix} = \begin{bmatrix} a_2 & b_2 & \frac{c_2}{f} \end{bmatrix} \end{aligned} \tag{4}$$

Since the angles between the normal vectors in the world frame and camera frame, respectively, are the same, then

$$\cos\alpha = \frac{\boldsymbol{n_{c1}}\cdot\boldsymbol{n}_{c2}^T}{\|\boldsymbol{n_{c1}}\|\|\boldsymbol{n}_{c2}\|} = \frac{a_1a_2 + b_1b_2 + \frac{c_1c_2}{f^2}}{\sqrt{a_1^2 + b_1^2 + \frac{c_1^2}{f^2}}\sqrt{a_2^2 + b_2^2 + \frac{c_2^2}{f^2}}} = \frac{m_1f^2 + m_2}{\sqrt{m_3f^2 + c_1^2}\sqrt{m_4f^2 + c_2^2}} \quad (5)$$

Let $\cos\alpha = m_5$ and then we can obtain a new equation

$$\left(m_3m_4m_5^2 - m_1^2\right)f^4 + \left(m_3m_5^2c_2^2 + m_4m_5^2c_1^2 - 2m_1m_2\right)f^2 + m_5^2c_1^2c_2^2 - m_2^2 = 0 \quad (6)$$

In Equation (6), f^2 is regarded as a parameter, and the above equation is a quadratic equation of one variable f^2 with two solutions. According to the restriction that $f > 0$ and $f^2 > 0$, the unique solution of focal length can be obtained. Now, the focal length estimation is finished. Then, the unit normal vectors in the camera frame can be given using

$$\begin{aligned} \boldsymbol{n_{c1}} &= \frac{\boldsymbol{n_{c1}}}{\|\boldsymbol{n_{c1}}\|} \\ \boldsymbol{n_{c2}} &= \frac{\boldsymbol{n_{c2}}}{\|\boldsymbol{n_{c2}}\|} \end{aligned} \quad (7)$$

Next, we will estimate the camera pose.

2.2. Pose Estimation

As can be seen from Section 2.1, the unit normal vectors $\boldsymbol{n}_{w1}, \boldsymbol{n}_{w2}, \boldsymbol{n}_{c1}, \boldsymbol{n}_{c2}$ are all known at present, and then we can obtain

$$\begin{aligned} \boldsymbol{n}_{c1} &= \boldsymbol{R}\cdot\boldsymbol{n}_{w1} \\ \boldsymbol{n}_{c2} &= \boldsymbol{R}\cdot\boldsymbol{n}_{w2} \end{aligned} \quad (8)$$

Here, $\boldsymbol{R}$ is the rotation matrix between the world frame and camera frame. According to the camera projection relationship, a point P_w in the world frame can be transformed to a point P_c in the camera frame using

$$P_c = \boldsymbol{R}\cdot P_w + \boldsymbol{t} \quad (9)$$

Here, t is the translation vector from the world frame to the camera frame. If we translate the origin of the world frame to the origin of the camera frame, t is zero and then

$$P_c = \boldsymbol{R}\cdot P_w \quad (10)$$

This is consistent with Equation (8) between the unit normal vectors. Inspired by this relationship, this paper regards the rotation relationship between the unit normal vectors as the rotation relationship between the points after the origins of the camera frame and the world frame coincide, and the coordinates of the corresponding points are the values of the unit normal vectors in the same frame, as shown in Figure 2.

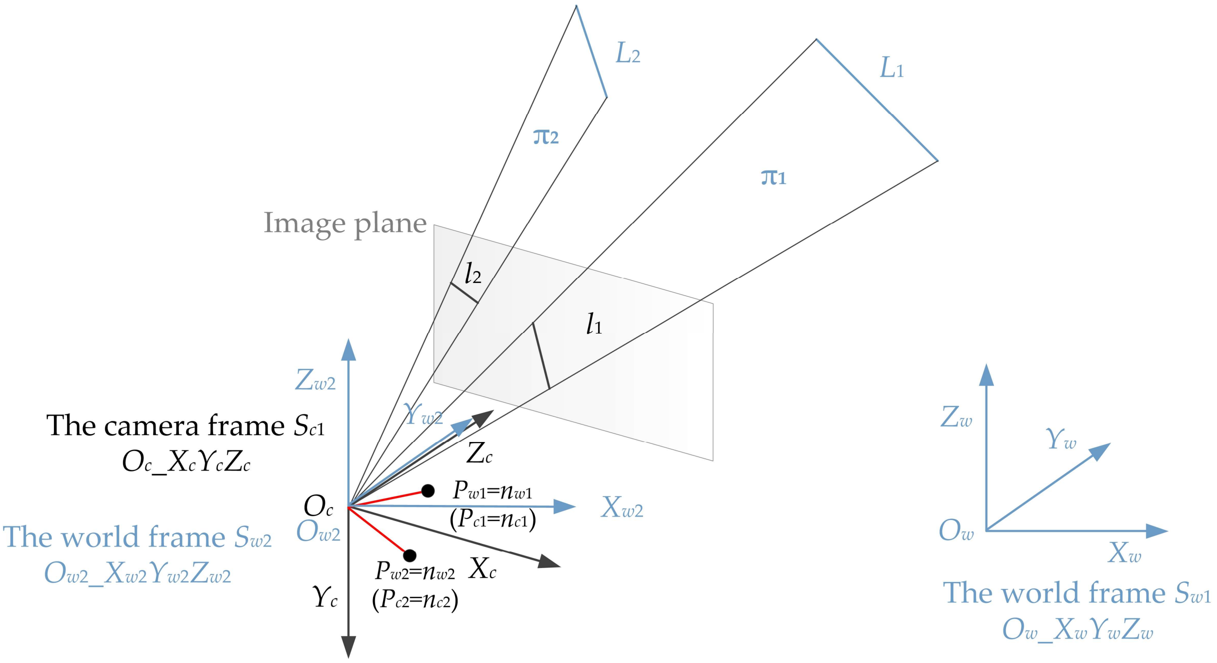

Figure 2. The geometry of our proposed method for pose estimation.

Here, a new world frame S_{w2} ($O_{w2}\_X_{w2}Y_{w2}Z_{w2}$) is established after translating the origin of the original world frame S_{w1} ($O_w\_X_wY_wZ_w$) to the origin of the camera frame S_{c1} ($O_c\_X_cY_cZ_c$). The relationship between the two world frames is as follows.

$$S_{w2} = S_{w1} - O_c \tag{11}$$

Now, the two points P_{c1} and P_{c2} in the camera frame S_{c1} are identical to the two points P_{w1} and P_{w2} in the world frame S_{w2}. Then, we establish another camera frame S_{c2} ($O_{c2}\_X_{c2}Y_{c2}Z_{c2}$) and world frame S_{w3} ($O_{w3}\_X_{w3}Y_{w3}Z_{w3}$), as shown in Figure 3.

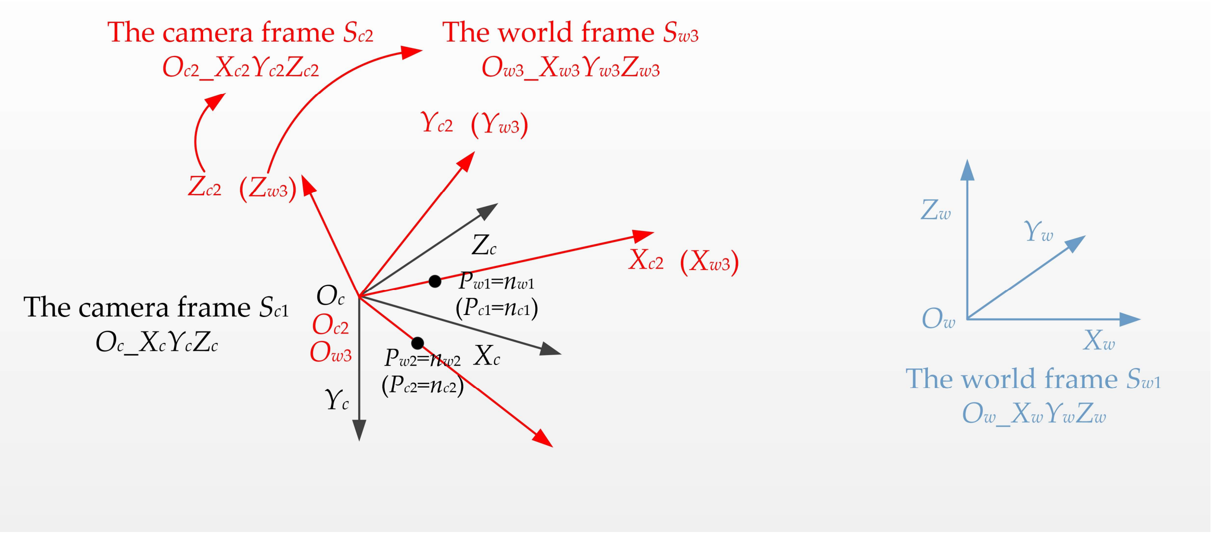

Figure 3. New camera frame and world frame plotted in red.

The origin O_{c2} of the new camera frame S_{c2} ($O_{c2}\_X_{c2}Y_{c2}Z_{c2}$) is located at the camera position O_c, and the unit direction vectors of each axis are calculated using

$$\begin{aligned}
\overrightarrow{O_{c2}X_{c2}} &= \frac{\overrightarrow{O_cP_{c1}}}{\|\overrightarrow{O_cP_{c1}}\|} \\
\overrightarrow{O_{c2}Z_{c2}} &= \frac{\overrightarrow{O_{c2}X_{c2}} \times \overrightarrow{O_cP_{c2}}}{\|\overrightarrow{O_{c2}X_{c2}} \times \overrightarrow{O_cP_{c2}}\|} \\
\overrightarrow{O_{c2}Y_{c2}} &= \overrightarrow{O_{c2}Z_{c2}} \times \overrightarrow{O_{c2}X_{c2}}
\end{aligned} \tag{12}$$

Then, the camera frame S_{c1} can be transformed into the camera frame S_{c2} using

$$\begin{aligned}
&S_{c2} = \boldsymbol{T}_{c\_c2} \cdot S_{c1} \\
&\boldsymbol{T}_{c\_c2} = \begin{bmatrix} \overrightarrow{O_{c2}X_{c2}} & \overrightarrow{O_{c2}Y_{c2}} & \overrightarrow{O_{c2}Z_{c2}} \end{bmatrix}^T
\end{aligned} \tag{13}$$

Similarly, we establish the new world frame S_{w3} ($O_{w3}\_X_{w3}Y_{w3}Z_{w3}$), and the origin of the world frame O_{w3} is also located at the camera position O_c. The unit direction vectors of each axis are calculated using

$$\begin{aligned}
\overrightarrow{O_{w3}X_{w3}} &= \frac{\overrightarrow{O_cP_{w1}}}{\|\overrightarrow{O_cP_{w1}}\|} \\
\overrightarrow{O_{w3}Z_{w3}} &= \frac{\overrightarrow{O_{w3}X_{w3}} \times \overrightarrow{O_cP_{w2}}}{\|\overrightarrow{O_{w3}X_{w3}} \times \overrightarrow{O_cP_{w2}}\|} \\
\overrightarrow{O_{w3}Y_{w3}} &= \overrightarrow{O_{w3}Z_{w3}} \times \overrightarrow{O_{w3}X_{w3}}
\end{aligned} \tag{14}$$

Then, the world frame S_{w2} can be transformed into the world frame S_{w3} using

$$\begin{aligned}
&S_{w3} = \boldsymbol{T}_{w2\_w3} \cdot S_{w2} \\
&\boldsymbol{T}_{w2\_w3} = \begin{bmatrix} \overrightarrow{O_{w3}X_{w3}} & \overrightarrow{O_{w3}Y_{w3}} & \overrightarrow{O_{w3}Z_{w3}} \end{bmatrix}^T
\end{aligned} \tag{15}$$

Since the two points P_{c1} and P_{c2} in the camera frame are the same as the two points P_{w1} and P_{w2} in the world frame, camera frame S_{c2} ($O_{c2}\_X_{c2}Y_{c2}Z_{c2}$) and world frame S_{w3} ($O_{w3}\_X_{w3}Y_{w3}Z_{w3}$) are the same frame. Now, the transformations between all the frames are determined, as shown in Figure 4.

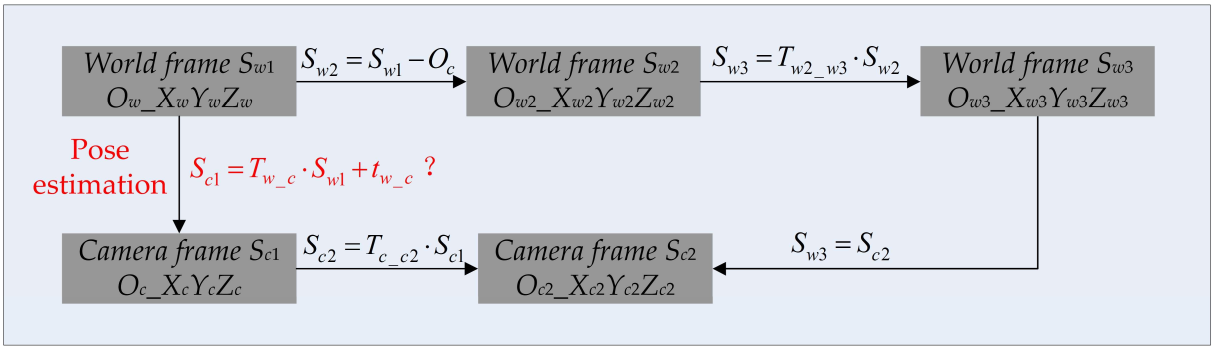
World frame S_{w1} $O_w\_X_wY_wZ_w$
$S_{w2} = S_{w1} - O_c$
World frame S_{w2} $O_{w2}\_X_{w2}Y_{w2}Z_{w2}$
$S_{w3} = T_{w2\_w3} \cdot S_{w2}$
World frame S_{w3} $O_{w3}\_X_{w3}Y_{w3}Z_{w3}$
Pose estimation
$S_{c1} = T_{w\_c} \cdot S_{w1} + t_{w\_c}$?
Camera frame S_{c1} $O_c\_X_cY_cZ_c$
$S_{c2} = T_{c\_c2} \cdot S_{c1}$
Camera frame S_{c2} $O_{c2}\_X_{c2}Y_{c2}Z_{c2}$
$S_{w3} = S_{c2}$

Figure 4. The transformations between each pair of frames. The transformation written in red is what needs to be solved in this paper, and this transformation contains all the pose information.

From Figure 4, we can obtain the relationship between the camera frame S_{c1} and the world frame S_{w1} using

$$\begin{aligned}
S_{c1} &= \boldsymbol{T}_{w\_c} \cdot S_{w1} + \boldsymbol{t}_{w\_c} \\
\boldsymbol{T}_{w\_c} &= \boldsymbol{T}_{c\_c2}^{-1} \cdot \boldsymbol{T}_{w2\_w3} \\
\boldsymbol{t}_{w\_c} &= -\boldsymbol{T}_{c\_c2}^{-1} \cdot \boldsymbol{T}_{w2\_w3} \cdot O_c
\end{aligned} \tag{16}$$

Now, the pose estimation is finished.

3. Experiments and Results

First, the proposed method and some other existing state-of-the-art solvers (i.e., P3P [16], RPnP [13], GPnPf [47], DLT [48] and P3L [24] here) with excellent performance are comprehensively tested with synthetic data, including numerical stability, noise sensitivity and computational speed, in terms of the focal length and pose estimation.

Second, the proposed method in this paper uses the prior knowledge of the camera position, but the prior knowledge cannot be absolutely correct, and an error in the camera position may bring a large error to the final pose and focal length estimation. Therefore, here, we also need to analyze the robustness of the pose and focal length estimation method to camera position noise.

Finally, the proposed method and some other existing state-of-the-art solvers are indirectly tested with real images, which shows that the proposed method can also achieve good performance in the real scenarios.

3.1. Synthetic Data

We synthesize a virtual perspective camera with 1280 × 800 resolution. Its principal point is the center of the image, the pixel size is 14 μm, and the focal length is 50 mm. The camera is located at (2, 2, 2) in meters in the world frame. All the methods are tested on synthetic data with a perfect pinhole camera to ensure that the comparison in computational performance is fair. For our proposed method and the P3L solver, 2D–3D line correspondences are needed, and for the P3P, RPnP, GPnPf and DLT solvers, 2D–3D point correspondences are needed. Hence, 3D points and lines are placed in the FOV of the virtual camera. In this paper, three thousand 3D points and three thousand 3D lines (their lengths are all 5 m) are randomly generated in a box of $[(-20, 20) \times (-20, 20) \times (180, 220)$ in meters in the world frame. Through the virtual camera, three thousand 2D points and three thousand 2D lines are generated. Then, we obtain the synthetic data, consisting of three thousand 2D–3D point correspondences and three thousand 2D–3D line correspondences.

In this section, two 2D–3D line correspondences and three 2D–3D line correspondences from the synthetic data are randomly chosen for our proposed method and the P3L solver, respectively, for each trial. Three, four, five and six 2D–3D point correspondences from the synthetic data are randomly chosen for the P3P, GPnPf, RPnP and DLT solvers for each trial, respectively. Here, P3P, GPnPf, RPnP, DLT and P3L are used to estimate the pose that includes the orientation and translation, and the GPnPf is also used to estimate the focal length when we analyze the performance of our proposed method.

3.1.1. Robustness to Camera Position Noise

The camera position is given by the RTK or total station in this paper, where the accuracy of RTK is better than 3 cm, and the accuracy of total station is better than 0.5 cm [49]. Therefore, the camera position has noise, which may affect the pose estimation. In this section, we need to analyze the robustness to the camera position noise, and then Gaussian noise, whose deviation level varies from 0 to 3 cm, is added to the camera position. Here, 10,000 random trials for each noise level are conducted independently, and the mean errors of rotation, translation, reprojection and focal length for each noise level are reported in Figure 5.

From Figure 5, it can be seen that as the camera position noise increases, so do the rotation error, the translation error, the reprojection error and the relative focal length. When the noise is 3 cm, the errors reach the maximum values, which are 0.08 degrees, 2.72 cm, 0.28 pixels and 0.008%, respectively. The results show that even if there are errors in rotation, translation, reprojection and focal length caused by camera position noise, they are still small, and we can say that our proposed method has strong robustness to camera position noise in terms of pose estimation.

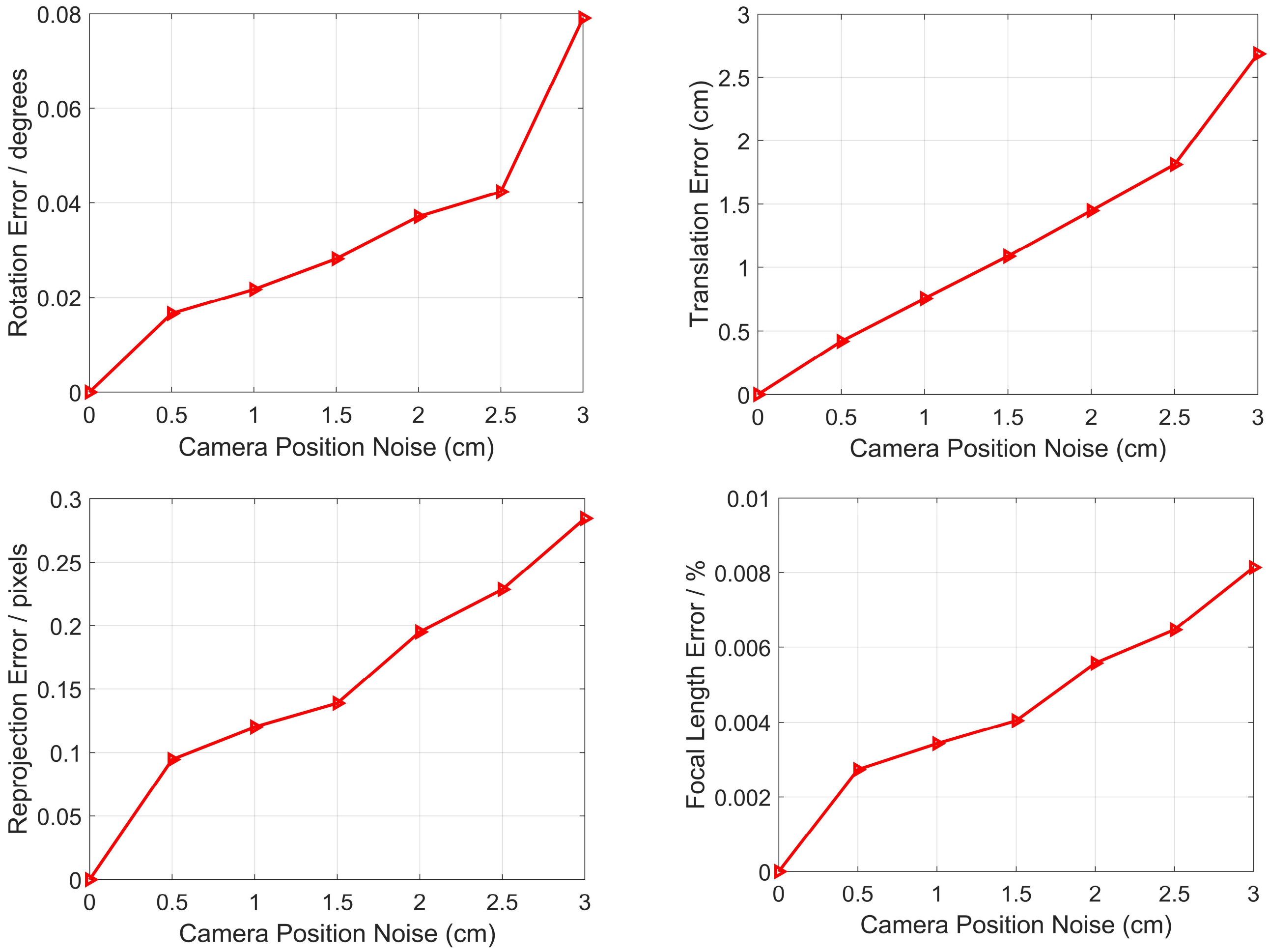

Figure 5. Robustness to camera position noise for the rotation error (**top left**), the translation error (**top right**), the reprojection error (**bottom left**) and the relative focal length (**bottom right**).

3.1.2. Numerical Stability

In this section, we test our proposed method in terms of numerical stability for pose estimation. In total, 10,000 independent trials are conducted using synthetic data with no noise added. We also compare our proposed method with P3P, RPnP, GPnPf, DLT and P3L, and the results are reported in Figure 6.

From Figure 6, it can be seen that all six methods have good numerical stability for pose estimation. Specifically, in terms of the rotation error, DLT obtains the best result, and our proposed method has the second. In terms of the translation error, our proposed method achieves the best and DLT achieve the second-best result. In terms of the reprojection error, P3P obtains the best result, RPnP obtains the second, DLT obtains the third and our proposed method obtains the fourth. As a whole, DLT has the best performance and our proposed method has the second-best performance for rotation, translation and reprojection.

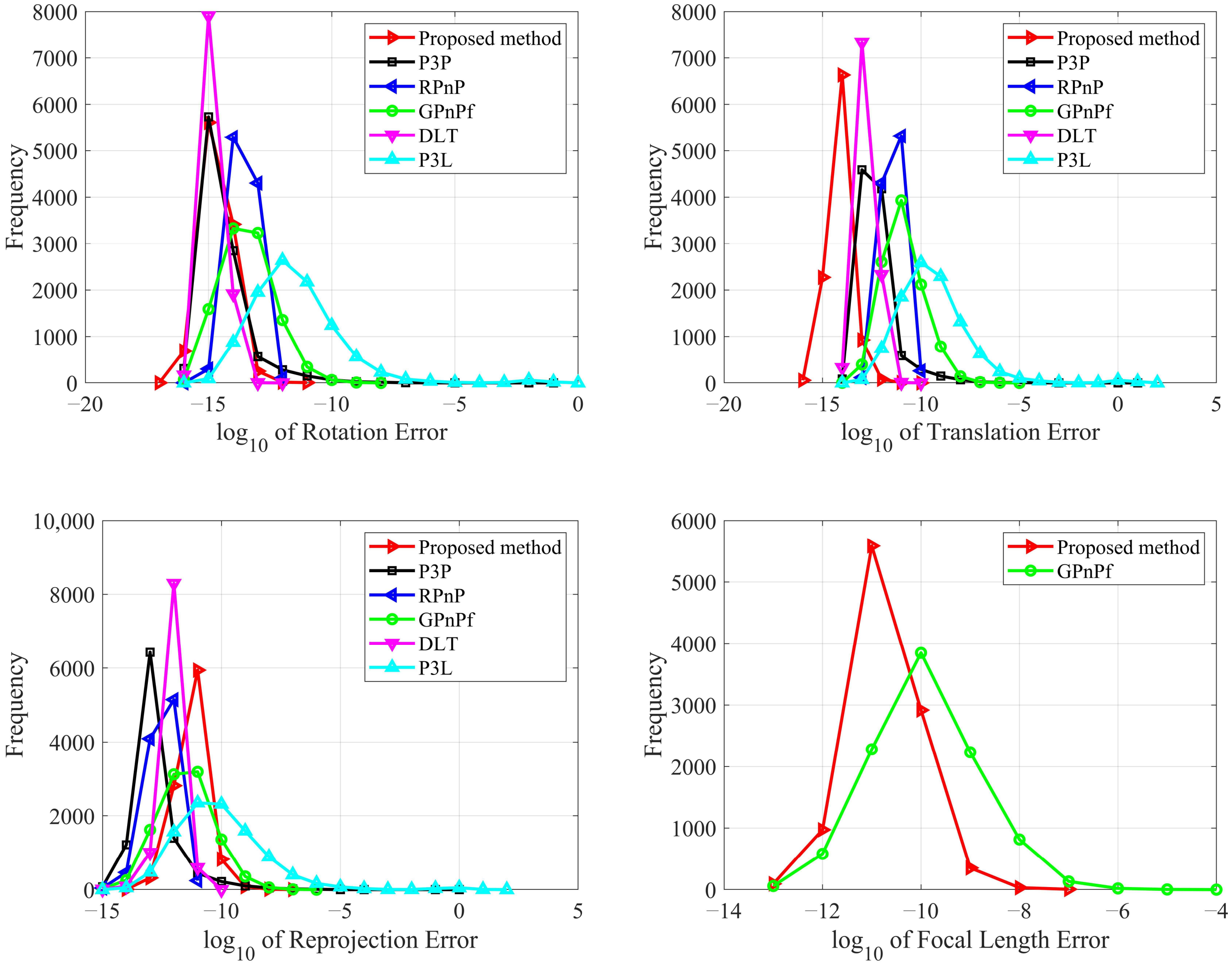

Figure 6. Numerical stability for our proposed method (Red), P3P (Black), RPnP (Blue), GPnPf (Green), DLT (Purple) and P3L (Cyan). The top left is the rotation error, the top right is the translation error, the bottom left is the reprojection error, and the bottom right is the focal length error.

Our proposed method also estimates the focal length, but the P3P, RPnP, DLT and P3L methods do not. Here, to analyze the numerical stability of the focal length estimation, we tested our proposed method compared with the GPnPf solver, which is one of the state-of-the-art solvers used to estimate the focal length, as shown in Figure 6 (bottom right). It can be seen that our proposed method and the GPnPf method both have good numerical stability, and our proposed method performs better than GPnPf. This result shows that our proposed method has better performance regarding numerical stability in terms of focal length estimation.

3.1.3. Noise Sensitivity

In this section, we test our proposed method in terms of noise sensitivity using synthetic data with zero-mean Gaussian noise added onto the 2D points and lines. The noise deviation level varies from 0 to 1 pixel and 10,000 independent trials are conducted for each noise level. We also compare our proposed method with P3P, RPnP, GPnPf, DLT and P3L, and the mean errors of rotation, translation, reprojection and focal length are reported in Figure 7.

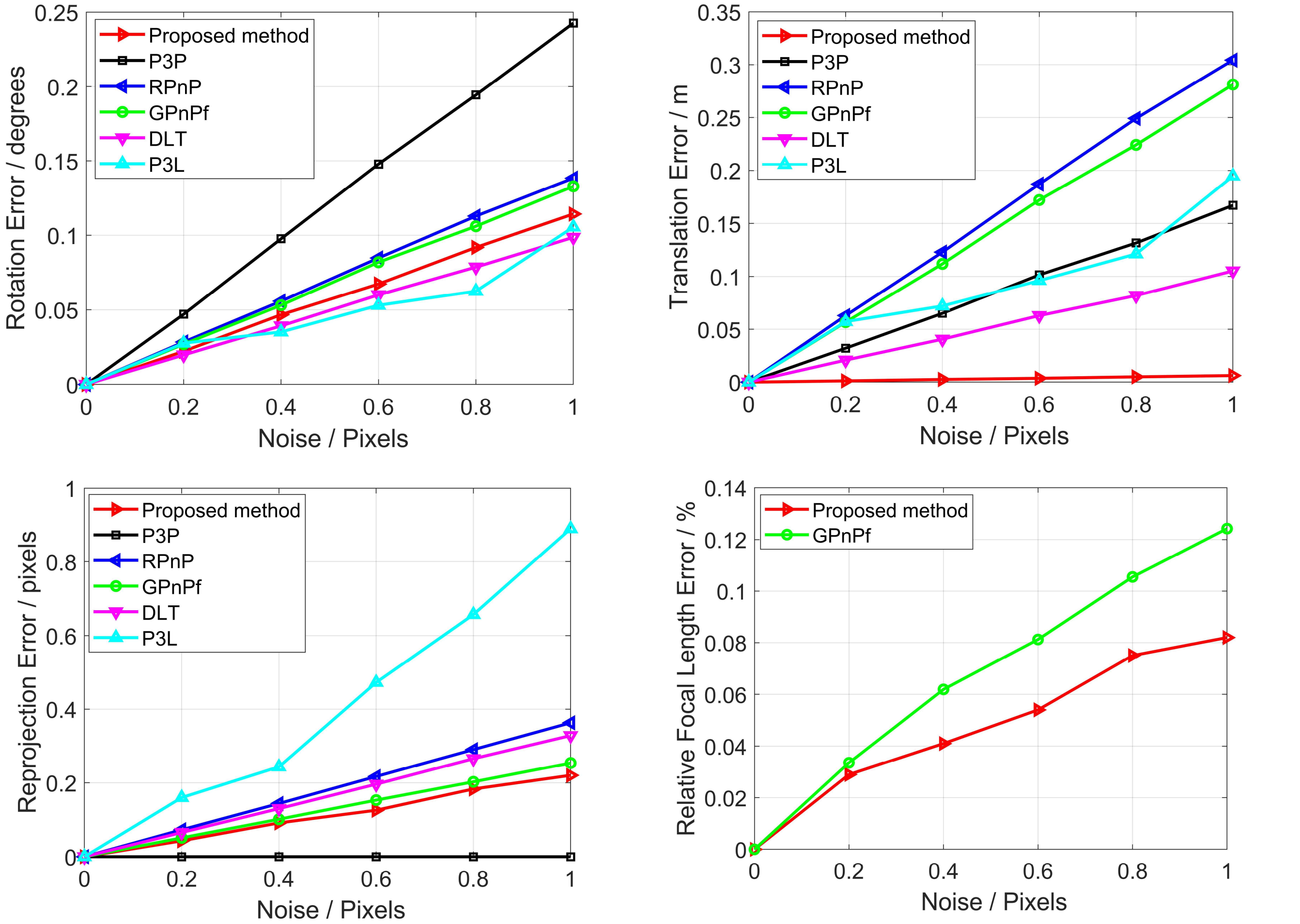

Figure 7. Noise sensitivity for our proposed method (Red), P3P (Black), RPnP (Blue), GPnPf (Green), DLT (Purple) and P3L (Cyan). The top left is the rotation error, the top right is the translation error, the bottom left is the reprojection error, and the bottom right is the focal length error.

From Figure 7, it can be seen that all six methods have good noise sensitivity for pose estimation. Moreover, as the noise increases, so do the rotation error, the translation error, the reprojection error and the focal length error. Specifically, in terms of the rotation error, our proposed method, DLT and P3L perform similarly, and they achieve the best performance. GPnPf shows the second-best result. In terms of the translation error, our proposed method is the best and DLT is the second-best. In terms of the reprojection error, P3P has the best result, our proposed method has the second-best, and DLT is fourth. As a whole, our proposed method has the best performance for rotation, translation and reprojection.

Our proposed method also estimates the focal length, but the P3P, RPnP, DLT and P3L methods do not. Here, to analyze the noise sensitivity of focal length estimation, we tested our proposed method compared with the GPnPf solver, which is one of the state-of-the-art solvers used to estimate the focal length, as shown in Figure 7 (bottom right). It can be seen that our proposed method and the GPnPf method both have good noise sensitivity, and our proposed method performs better than GPnPf. This result shows that our proposed method has better performance regarding noise sensitivity in terms of focal length estimation.

3.1.4. Computational Speed

Sections 3.1.1–3.1.3 show that our proposed method has good performance in terms of accuracy and stability. However, to fully evaluate a method, it is necessary to test the computational speed, because some methods achieve better results only when they use more complex computational processes, at the expense of computational speed. Therefore, we need to demonstrate the complexity of our proposed method by evaluating the computational speed.

In this section, 10,000 independent trials are conducted on a 3.3 GHz two-core laptop for all the methods, respectively, to test the computational speed. Then, the mean computational times are calculated, as reported in Table 1.

Table 1. Computational time.

Method	Our Proposed Method	P3P	RPnP	GPnPf	DLT	P3L
Computational time	0.46 ms	1.72 ms	2.03 ms	6.88 ms	0.77 ms	1.56 ms

From Table 1, it can be seen that our proposed method has the fastest computational speed according to the computational time. Specifically, the computational speed of our proposed method is 3.7 times, 4.4 times 15.0 times, 1.7 times and 3.4 times that of the latter five methods, respectively.

3.2. Real Images

In this section, our proposed method is indirectly tested with real images to show that it can work well with real scenarios. Simultaneously, we compare the performance of our proposed method with that of the other methods (i.e., P3P, RPnP, GPnPf, DLT and P3L solvers).

In real scenarios, the ground truth of the camera pose is unknown and hence we cannot directly compare the value of the pose estimation with the ground truth. However, the positions of the spatial point and line are easily measured with high accuracy using some common tools (e.g., RTK, total station and ruler). Hence, the position can be used to test our proposed method indirectly in real images. In fact, pose estimation is preparation for practical applications, such as SfM, SLAM and photogrammetry. One important application is to measure the spatial positions of objects in photogrammetry, where two or more cameras after pose estimation are used to finish the measurement, namely stereo vision. It can be seen that the accuracy of the position of the point is affected by the pose estimation and, hence, the accuracy of the position can reflect the accuracy of pose estimation.

Here, two rectangular boxes are placed on a checkerboard, whose sizes are known. Many points and lines exist in this scenario, and their positions are all known as ground truth. Two mobile phone cameras are used to capture real images from two different perspectives, as shown in Figure 8.

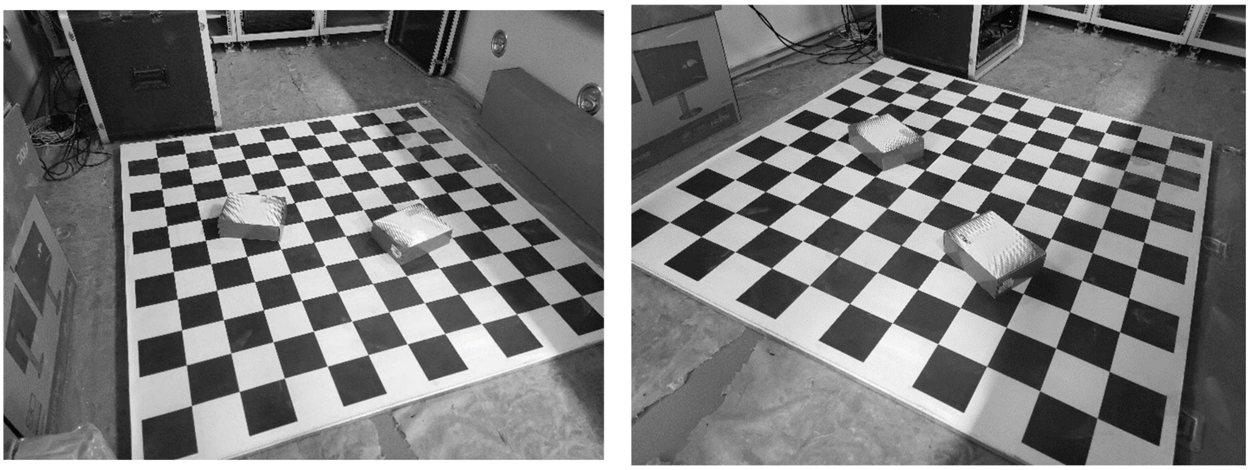

Figure 8. Real images captured from two different perspectives. Two rectangular boxes are placed on a checkerboard; the sizes of the boxes and checkerboard are known.

The focal length of the camera is 4.71 mm, the resolution is 4000 × 3000, and the pixel size is 1.6µm. The positions of the two cameras are given by the total station. Then, two lines are randomly chosen to estimate the pose for our proposed method, three lines for the P3L method, three points for the P3P method, four points for the GPnPf method, five points for the RPnP method and six points for the DLT method. After pose estimation using the six methods, the positions of the remaining points can be given as measured values by stereo vision [50]. Here, the measured values are denoted as P_i, $(1 \leq i \leq n)$, where n is the number of remaining points for the checkerboard. Because the size of the checkerboard is known, the positions of these points are known as the ground truths, which are denoted as P_i'. We use the mean relative position errors $E_{position}$, between the measured value and the ground truth, to indicate the accuracy of all six methods, and the formula is written as

$$E_{position} = \frac{\sum\limits_{i=1}^{n} \frac{\left|P_i - P_i'\right|}{P_i'}}{n} \tag{17}$$

The mean relative position errors between the measured value and the ground truth are reported in Table 2.

Table 2. Relative errors of position.

Method	Our Proposed Method	P3P	RPnP	GPnPf	DLT	P3L
Mean relative error %	0.43	0.51	1.79	1.06	0.56	0.62

From Table 2, it can be seen that our proposed method has the lowest error, the P3P method has the second-lowest and the RPnP method has the highest. This is consistent with the results for the synthetic data and shows that our proposed method can work well with synthetic data and real images.

In addition, we also test our proposed method in terms of computational speed in real images, and the experiment is conducted on a 3.3 GHz two-core laptop for all the methods. Specifically, the mean computational speed of our proposed method is 3.4 times, 3.9 times 16.2 times, 1.5 times and 3.7 times that of the latter five methods, respectively. This is basically consistent with the results for the synthetic data.

Last, we compare the projection of the known 3D line using the estimated pose to the corresponding line segment on the real images, and then we can intuitively observe whether they match, as shown in Figure 9.

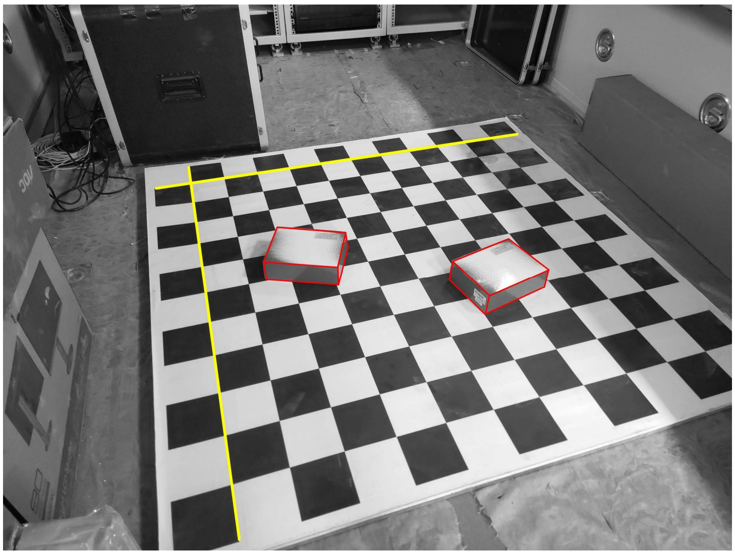

Figure 9. The projections (Red) of the known lines using the estimated pose with two 3D lines (Yellow).

The projection is affected by the pose estimation, and, hence, the error between the projection and the corresponding line segment on the real images can intuitively reflect the error of pose estimation. From Figure 9, it can be seen that the error between the projection and the corresponding line segment on the real images is low, which shows, from another perspective, that our proposed method has good performance in real scenarios.

4. Discussion

In this paper, we propose a fast and accurate method to estimate the focal length and pose using two 2D–3D line correspondences and the camera position. Our core contribution is to convert the PnL problem with 2D–3D line correspondences into the estimation problem with 3D–3D point correspondences. To the best of our knowledge, this is the first study to use the camera position and line correspondences simultaneously to estimate the focal length and pose. Using the camera position as the prior knowledge, two planes can be obtained in the camera frame and world frame, respectively. Then, two angles between the two planes in the camera frame and world frame, respectively, can be obtained. Since there is only a rigid body transformation, the angle remains the same in either frame. Hence, the two angles are equal, and, using this information, we can estimate the focal length efficiently and independently.

When we estimate the camera pose, another geometric characteristic is used, where we establish the transform between the unit normal vectors of the two planes with this characteristic, and this transform can be regarded as the camera projection of the 3D point. Then, the pose estimation using line correspondences is converted into pose estimation using 3D–3D point correspondences in intermediate frames, and the latter pose estimation can be finished quickly using traditional point-based solvers. The differences and advantages of our proposed method are discussed as follows.

4.1. Differences and Advantages

Compared with other existing methods using 2D–3D line correspondences, the first difference is to use the camera position as prior knowledge, which can simplify the problem of the focal length and pose estimation. The known camera position can reduce the number of 2D–3D line correspondences, improve the efficiency and expand the applicability. The real scene might not have sufficient line correspondences, and, in this case, our proposed method, with fewer 2D–3D line correspondences, can work better with the scenario. In addition, if the scene has mass line correspondences and outliers exist, generally, the RANSAC (RANdom SAmple Consensus) is used to eliminate the outliers. The RANSAC has a key parameter, which is the minimal set of correspondences for estimating pose. The smaller the minimal set, the faster the computational speed of eliminating the outliers. Hence, the RANSAC with our proposed method is faster than that with other existing methods when the scene has mass line correspondences and outliers exist.

When the focal length is estimated, only a quadratic equation of one variable is used. We can solve the focal length directly, not using iterations and nonlinear algorithms, which is the main reason that our proposed method has better numerical stability and noise sensitivity in terms of focal length estimation. In addition, using the characteristic whereby the focal length is greater than zero, a single solution can be obtained, and unlike some existing methods, no other constraint is needed. The two characteristics described above are one reason that our proposed method has faster computational speed. When the camera pose is estimated, the main calculation process only involves multiplication, division and matrix operation, and there is also no nonlinear computation. This is the main reason that our proposed method has good performance in terms of numerical stability, noise sensitivity and computational speed in terms of pose estimation.

In addition, because our proposed method does not involve nonlinear computation and iteration, there is no multi-solution phenomenon. This is another reason that our proposed method performs better than the other existing methods in terms of computational speed, as described in Section 3.1.4. It can be also seen that our proposed method and the

DLT solver perform best in terms of numerical stability, noise sensitivity and computational speed compared to the other four methods. The main reason is that our proposed method and the DLT solver do not both involve nonlinear computation, but the other four methods do.

The level of simplification using prior knowledge is different, which means that the benefit might be high or low for different methods. As described in Section 3.1.1, our proposed method has good robustness to camera position noise, and this means that any error in the camera position does not bring a significant error for focal length and pose estimation. This is also a reason that our proposed method has higher accuracy.

Briefly, the proposed method has the following differences and advantages. (1) It is the first method to use the camera position and only two 2D–3D line correspondences simultaneously to estimate the focal length and pose; (2) it has strong robustness to prior knowledge, i.e., the known camera position; (3) no multi-solution phenomenon exists both for focal length and pose estimation; (4) it has better performance in terms of numerical stability, noise sensitivity and computational speed; (5) it can work well with synthetic data and real scenarios. However, the main disadvantage is that the camera position needs to be known in advance and many cases might not have the necessary conditions to measure the camera position.

4.2. Future Work

In the future, we will extend the idea in this paper to continue two main tasks. First, we will use more 2D–3D line correspondences and known camera positions to estimate more intrinsic parameters (e.g., radial distortion), rather than only the focal length and pose, which will expand the usable range of our proposed method. Second, as described in Section 4.1, the RANSAC + our proposed method will be used to estimate the camera pose from mass 2D–3D line correspondences, even if the line correspondences have some outliers. Our ultimate goal is to use RANSAC + our proposed method to work with real scenarios, e.g., SfM and SLAM. In addition, our method and the other methods in this paper used good lines. If imperfect lines exist, we could use the combination of a line detection algorithm and manual intervention to further extract and optimize the imperfect lines. This might be done in the future.

5. Conclusions

We proposed a fast and accurate method to estimate the focal length and pose using only two 2D–3D line correspondences and the camera position. The geometric characteristic whereby the angle between two planes is not changed in different frames is used to estimate the focal length. Then, another geometric characteristic, whereby the pose estimation using 2D–3D line correspondences in this paper can be converted into pose estimation using 3D–3D point correspondences in intermediate frames, is used to estimate the pose. The two calculation processes do not involve nonlinear computation and both have no multi-solution phenomenon. Experimental results show that our proposed method has better performance in terms of numerical stability, noise sensitivity and computational speed in synthetic data and real images compared to several state-of-the-art pose estimation solvers.

Author Contributions: Conceptualization, K.G. and Z.Z. (Zhixiang Zhang); methodology, K.G.; software, K.G. and Z.Z. (Zhongsen Zhang); validation, Y.T.; formal analysis, K.G. and H.C.; investigation, Z.Z. (Zhongsen Zhang); resources, K.G.; data curation, Y.T.; writing—original draft preparation, Z.Z. (Zhixiang Zhang); writing—review and editing, K.G.; visualization, K.G.; supervision, H.C.; project administration, Z.Z. (Zhongsen Zhang); funding acquisition, Z.Z. (Zhixiang Zhang). All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available in the manuscript.

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

References

1. Ekanayake, B.; Wong, J.K.W.; Fini, A.A.F.; Smith, P. Computer vision-based interior construction progress monitoring: A literature review and future research directions. *Autom. Constr.* **2021**, *127*, 103705. [CrossRef]
2. Yu, Q.; Xu, G.; Wang, Z.; Li, Z. An efficient and globally optimal solution to perspective-n-line problem. *Chin. J. Aeronaut.* **2022**, *35*, 400–407. [CrossRef]
3. Zhou, F.; Cui, Y.; Wang, Y.; Liu, L.; Gao, H. Accurate and robust estimation of camera parameters using RANSAC. *Opt. Lasers Eng.* **2013**, *51*, 197–212. [CrossRef]
4. Ding, Y.; Hua, L.; Li, S. Research on computer vision enhancement in intelligent robot based on machine learning and deep learning. *Neural Comput. Appl.* **2022**, *34*, 2623–2635. [CrossRef]
5. Mikhail, E.M.; Bethel, J.S.; McGlone, J.C. *Introduction to Modern Photogrammetry*; John Wiley & Sons: New York, NY, USA, 2001.
6. Kendall, A.; Grimes, M.; Cipolla, R. Posenet: A convolutional network for real-time 6-dof camera relocalization. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 2938–2946.
7. Melekhov, I.; Ylioinas, J.; Kannala, J.; Rahtu, E. Relative camera pose estimation using convolutional neural networks. In Proceedings of the International Conference on Advanced Concepts for Intelligent Vision Systems, Antwerp, Belgium, 18–21 September 2017; pp. 675–687.
8. Sattler, T.; Zhou, Q.; Pollefeys, M.; Leal-Taixe, L. Understanding the limitations of cnn-based absolute camera pose regression. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 3302–3312.
9. En, S.; Lechervy, A.; Jurie, F. Rpnet: An end-to-end network for relative camera pose estimation. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–11 September 2018; pp. 738–745.
10. Guo, K.; Ye, H.; Gu, J.; Chen, H. A Novel Method for Intrinsic and Extrinsic Parameters Estimation by Solving Perspective-Three-Point Problem with Known Camera Position. *Appl. Sci.* **2021**, *11*, 6014. [CrossRef]
11. Wu, Y.; Hu, Z. PnP problem revisited. *J. Math. Imaging Vis.* **2006**, *24*, 131–141. [CrossRef]
12. Zhou, L.; Yang, Y.; Abello, M.; Kaess, M. A robust and efficient algorithm for the PnL problem using algebraic distance to approximate the reprojection distance. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; pp. 9307–9315.
13. Li, S.; Xu, C.; Xie, M. A robust O (n) solution to the perspective-n-point problem. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 1444–1450. [CrossRef] [PubMed]
14. Wang, P.; Chou, Y.; An, A.; Xu, G. Solving the PnL problem using the hidden variable method: An accurate and efficient solution. *Vis. Comput.* **2022**, *38*, 95–106. [CrossRef]
15. Vakhitov, A.; Funke, J.; Moreno-Noguer, F. Accurate and linear time pose estimation from points and lines. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; pp. 583–599.
16. Kneip, L.; Scaramuzza, D.; Siegwart, R. A novel parametrization of the perspective-three-point problem for a direct compu-tation of absolute camera position and orientation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Colorado Springs, CO, USA, 20–25 June 2011; pp. 2969–2976.
17. Lee, G.H. A minimal solution for non-perspective pose estimation from line correspondences. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; pp. 170–185.
18. Zheng, Y.; Kuang, Y.; Sugimoto, S.; Astrom, K.; Okutomi, M. Revisiting the pnp problem: A fast, general and optimal solution. In Proceedings of the IEEE International Conference on Computer Vision, Sydney, Australia, 1–8 December 2013; pp. 2344–2351.
19. Josephson, K.; Byrod, M. Pose estimation with radial distortion and unknown focal length. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Miami Beach, FL, USA, 20–25 June 2009; pp. 2419–2426.
20. Gao, X.S.; Hou, X.R.; Tang, J.; Cheng, H.F. Complete solution classification for the perspective-three-point problem. *IEEE Trans. Pattern Anal. Mach. Intell.* **2003**, *25*, 930–943.
21. Zhang, Y.; Li, X.; Liu, H.; Shang, Y. Probabilistic approach for maximum likelihood estimation of pose using lines. *IET Comput. Vis.* **2016**, *10*, 475–482. [CrossRef]
22. Zhou, L.; Koppel, D.; Kaess, M. A complete, accurate and efficient solution for the perspective-n-line problem. *IEEE Robot. Autom. Lett.* **2020**, *6*, 699–706. [CrossRef]
23. Zhang, L.; Xu, C.; Lee, K.M.; Koch, R. Robust and efficient pose estimation from line correspondences. In Proceedings of the Asian Conference on Computer Vision, Daejeon, Korea, 5–9 November 2012; pp. 217–230.
24. Xu, C.; Zhang, L.; Cheng, L.; Koch, R. Pose estimation from line correspondences: A complete analysis and a series of solutions. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *39*, 1209–1222. [CrossRef] [PubMed]
25. Liu, C.; Zhu, F.; Jin, Z.; Xia, R. Pose determination from one point and two coplanar line features. In Proceedings of the International Conference on Intelligent Computing, Huangshan, China, 25–29 July 2012; pp. 577–584.
26. Guo, K.; Ye, H.; Gu, J.; Tian, Y. A Fast and Simple Method for Absolute Orientation Estimation Using a Single Vanishing Point. *Appl. Sci.* **2022**, *12*, 8295. [CrossRef]

27. Hartley, R.; Zisserman, A. *Multiple View Geometry in Computer Vision*, 2nd ed.; Cambridge University Press: Cambridge, UK, 2003.
28. Wolfe, W.; Mathis, D.; Sklair, C.; Magee, M. The perspective view of three points. *IEEE Trans. Pattern Anal. Mach. Intell.* **1991**, *13*, 66–73. [CrossRef]
29. Yin, X.; Ma, L.; Tan, X.; Qin, D. A Robust Visual Localization Method with Unknown Focal Length Camera. *IEEE Access* **2021**, *9*, 42896–42906. [CrossRef]
30. Penate-Sanchez, A.; Andrade-Cetto, J.; Moreno-Noguer, F. Exhaustive linearization for robust camera pose and focal length estimation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 2387–2400. [CrossRef]
31. Kukelova, Z.; Bujnak, M.; Pajdla, T. Real-time solution to the absolute pose problem with unknown radial distortion and focal length. In Proceedings of the IEEE International Conference on Computer Vision, Sydney, Australia, 1–8 December 2013; pp. 2816–2823.
32. Zhao, Z.; Ye, D.; Zhang, X.; Chen, G.; Zhang, B. Improved direct linear transformation for parameter decoupling in camera calibration. *Algorithms* **2016**, *9*, 31. [CrossRef]
33. Barone, F.; Marrazzo, M.; Oton, C.J. Camera Calibration with Weighted Direct Linear Transformation and Anisotropic Uncertainties of Image Control Points. *Sensors* **2020**, *20*, 1175. [CrossRef]
34. Kukelova, Z.; Bujnak, M.; Pajdla, T. Closed-form solutions to minimal absolute pose problems with known vertical direction. In Proceedings of the Asian Conference on Computer Vision, Queenstown, New Zealand, 8–12 November 2010; pp. 216–229.
35. Kalantari, M.; Hashemi, A.; Jung, F.; Gudeon, J.-P. A new solution to the relative orientation problem using only 3 points and the vertical direction. *J. Math. Imaging Vis.* **2011**, *39*, 259–268. [CrossRef]
36. D'Alfonso, L.; Garone, E.; Muraca, P.; Pugliese, P. On the use of IMUs in the PnP Problem. In Proceedings of the International Conference on Robotics and Automation, Hong Kong, China, 31 May–5 June 2014; pp. 914–919.
37. Guo, K.; Ye, H.; Chen, H.; Gao, X. A New Method for Absolute Pose Estimation with Unknown Focal Length and Radial Distortion. *Sensors* **2022**, *22*, 1841. [CrossRef]
38. Guo, K.; Ye, H.; Zhao, Z.; Gu, J. An Efficient Closed Form Solution to the Absolute Orientation Problem for Camera with Unknown Focal Length. *Sensors* **2021**, *21*, 6480. [CrossRef] [PubMed]
39. Wang, P.; Xu, G.; Cheng, Y. A novel algebraic solution to the perspective-three-line pose problem. *Comput. Vis. Image Underst.* **2020**, *191*, 102711. [CrossRef]
40. Abdellali, H.; Frohlich, R.; Kato, Z. A direct least-squares solution to multi-view absolute and relative pose from 2d–3d perspective line pairs. In Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, Seoul, Korea, 27–28 October 2019; pp. 2119–2128.
41. Abdellali, H.; Frohlich, R.; Kato, Z. Robust absolute and relative pose estimation of a central camera system from 2d–3d line correspondences. In Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, Seoul, Korea, 27–28 October 2019; pp. 895–904.
42. García Fontán, J.; Nayak, A.; Briot, S.; Safey El Din, M. Singularity Analysis for the Perspective-Four and Five-Line Problems. *Int. J. Comput. Vis.* **2022**, *130*, 909–932. [CrossRef]
43. Kuang, Y.; Astrom, K. Pose estimation with unknown focal length using points, directions and lines. In Proceedings of the IEEE International Conference on Computer Vision, Sydney, Australia, 1–8 December 2013; pp. 529–536.
44. Yu, Q.; Xu, G.; Cheng, Y. An efficient and globally optimal method for camera pose estimation using line features. *Mach. Vis. Appl.* **2020**, *31*, 1–11. [CrossRef]
45. Přibyl, B.; Zemčík, P.; Čadík, M. Absolute pose estimation from line correspondences using direct linear transformation. *Comput. Vis. Image Underst.* **2017**, *161*, 130–144. [CrossRef]
46. Lecrosnier, L.; Boutteau, R.; Vasseur, P.; Savatier, X.; Fraundorfer, F. Camera pose estimation based on PnL with a known vertical direction. *IEEE Robot. Autom. Lett.* **2019**, *4*, 3852–3859. [CrossRef]
47. Zheng, Y.; Sugimoto, S.; Sato, I.; Okutomi, M. A general and simple method for camera pose and focal length determination. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 430–437.
48. Zhang, Z. A flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell.* **2000**, *22*, 1330–1334. [CrossRef]
49. Guo, K.; Ye, H.; Gao, X.; Chen, H. An Accurate and Robust Method for Absolute Pose Estimation with UAV Using RANSAC. *Sensors* **2022**, *22*, 5925. [CrossRef]
50. Do, P.N.B.; Nguyen, Q.C. A review of stereo-photogrammetry method for 3-D reconstruction in computer vision. In Proceedings of the IEEE 19th International Symposium on Communications and Information Technologies, Ho Chi Minh City, Vietnam, 25–27 September 2019; pp. 138–143.