

Enhancing Sensor Measurements through Wide Baseline Stereo Images

Rimon Elias

Department of Computer Science and Engineering, German University in Cairo, Cairo, Egypt

Received 26 May 2008; accepted 18 June 2008

Abstract

In this paper, we suggest an algorithm to enhance the accuracy of sensor measurements representing camera parameters. The process proposed is based solely on a pair of wide baseline (or sparse view) images. We use the so-called JUDOCA operator to extract junctions. This operator produces junctions in terms of locations as well as orientations. Such an information is used to estimate an affine transformation matrix, which is used to guide a variance normalized correlation process that produces a set of possible matches. The fundamental matrix can be easily estimated using the so-called RANSAC scheme. Consequently, the essential matrix can be derived given the available calibration matrix. The essential matrix is then decomposed using Singular Value Decomposition. In addition to a translation vector, this decomposition results in a rotation matrix with accurate rotation angles involved. Mathematical derivation is done to extract angles from the rotation matrix and express them in terms of different rotation systems.

Key Words: Wide baseline matching, sparse view matching, parameters recovery, rotation systems, JUDOCA, junction detection, feature detection.

1 INTRODUCTION

Accurate information makes many Computer Vision tasks much easier. For example, point matching and 3D reconstruction of objects would be very much facilitated once the camera parameters for the observed scene are known accurately. Unfortunately, in many cases, sensors used to capture camera parameters provide measurements that may be characterized by uncertainty. Different types of errors may be present and vary from fixed errors to random errors [11]. Fixed errors (also known as *bias*) are constant deviations from the accurate value while random errors (also known as *precision*) vary over time. Such inaccuracy may be indicated as a fixed value and a range where the true value lies within (e.g., ± 10); or a percentage of reading (e.g., 0.5% of the reading). These types of uncertainty or error representation provide margins that should be taken into account if any computational processing is to be applied to the measurements read.

Fortunately, information embedded in the images captured can be used to tune the measurements read. Such a process may get easier in case of short baseline stereo images. In this case, the straightforward scheme to enhance the accuracy of the rotation angles is by getting a set of good matches and inferring the epipolar geometry accordingly. However, the task gets harder if the images captured are a pair of wide baseline images

Correspondence to: <rimon.elias@guc.edu.eg>

Recommended for acceptance by David Fofi and Ralph Seulin

ELCVIA ISSN:1577-5097

Published by Computer Vision Center / Universitat Autònoma de Barcelona, Barcelona, Spain

as perspective deformation imposes a considerable problem. In this paper, we want to tackle the problem of enhancing the accuracy of the extrinsic camera parameters; especially the three rotation angles, utilizing a pair of wide baseline images. As different systems can be used to express how a camera is rotated in order to capture two images for the same scene, we will present the enhanced rotation angles in two different rotation systems for easy use by different vision algorithms.

Aiming to produce a reliable solution, we suggest using the information inferred by detecting junctions as a primary step to rotation angle enhancement. Detecting junctions does not rely on the availability of camera parameters. Hence, the inaccuracy existing in camera parameters does not affect this detection step. On the contrary, junction detection can be utilized to add more accuracy to the parameters.

This work is not meant to be an intensive research project. Rather, it is mainly a sequence of previously defined methods we use to enhance the accuracy of rotation angles and can be integrated as a software component with commercial sensors.

The paper is organized as follows. Sec. 2 summarizes the steps of the algorithm proposed. Mathematical bases used to implement our algorithm are detailed in the subsequent sections (Sec. 3 to Sec. 7). Sec. 8 provides experimental results and discusses suggestions to test how accurate the parameters have become. Finally, Sec. 9 provides a final conclusion.

2 Algorithm to Enhance Angles Accuracy

Fig. 1 depicts the steps of the algorithm we propose to enhance the accuracy of the rotation angles between a pair of wide baseline images that are read by inaccurate sensors.

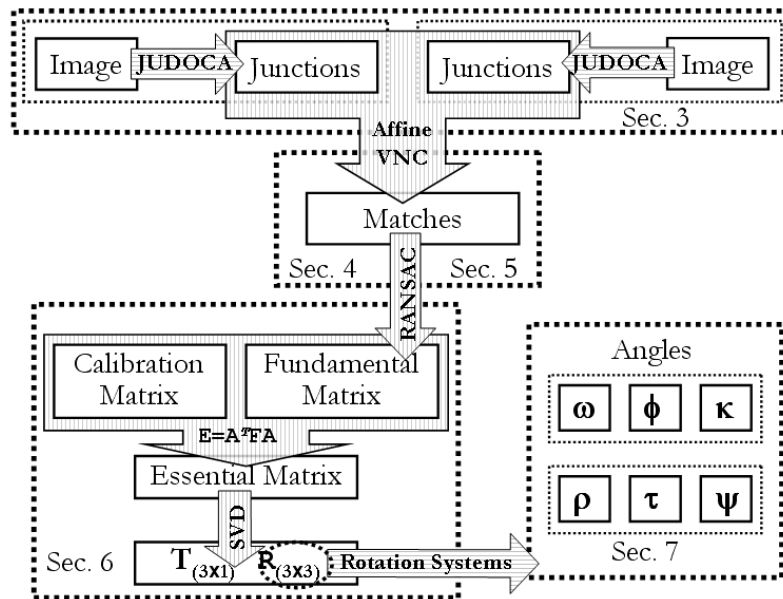


Figure 1: Steps of the proposed algorithm.

The steps can be summarized as follows:

1. Detect junctions in both images using the JUDOCA operator [9] (Sec. 3).
2. Use junction information to establish an affine transformation matrix that approximates the homography among corresponding junctions (Sec. 4).

3. Perform variance normalized correlation (VNC) after transformation. This step will result in a number of matches (Sec. 5).
4. Use RANSAC to estimate the fundamental matrix and exclude outliers (Sec. 6).
5. Use the calibration matrix to get the essential matrix (Sec. 6).
6. Decompose the essential matrix through Singular Value Decomposition (SVD). This provides the accurate rotation matrix and translation vector (Sec. 6).
7. The angles of rotation can be recovered from the rotation matrix using different rotation systems (Sec. 7).

The above steps will be explained in more detail below.

3 Detecting Junctions through JUDOCA

The JUDOCA operator [9] is a junction detector that may be used to detect not only the locations of junctions but their orientations as well (i.e., the inclination angles of the branches forming the junctions as well as the number of branches). The operator is based on determining the gradient magnitude through vertical and horizontal Gaussian filters according to some variance value. Fig. 2(b) shows the gradient magnitude of a square part of the image shown in Fig. 2(a).

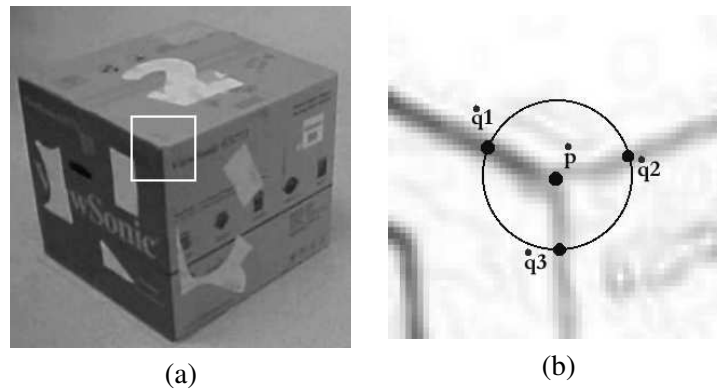


Figure 2: An example of a 3-edge Y-junction. (a) One corner of a box that produces a Y-junction. (b) A circular mask is centered at the position of the junction \dot{p} with three points on the circumference (superimposed on the gradient image).

The JUDOCA operator scans along the branches of the junctions to detect their existence; however, scanning throughout grayscale regions may be time consuming. In order to speed up the detection process, the operator creates two binary images; \mathcal{B} and \mathcal{B}^+ . The image \mathcal{B} is created by imposing a threshold on the gradient magnitude while \mathcal{B}^+ is created by calculating the local maxima.

Working on the first binary image \mathcal{B} , the operator uses a circular mask at every point and a list of points in \mathcal{B}^+ on the circumference of the mask is obtained. The operator scans the radial segments connecting the center of the mask and each of the circumferential points obtained previously. If the points scanned do not belong to \mathcal{B} , the junction is rejected. If the points belong to \mathcal{B}^+ , the junction strength is incremented.

In addition to its speed as working on binary images, this operator provides junctions of good accuracy in terms of their locations, number of edges forming each junction and their orientations as well. An example of detecting junctions applied to a real-world image is shown in Fig. 3. This operator is freely available online.*

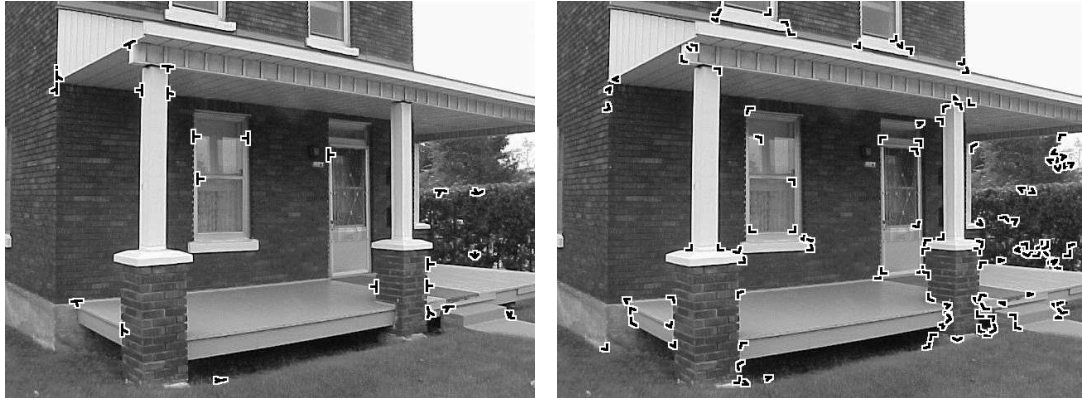


Figure 3: An example of detecting junctions using JUDOCA applied to a real-world image. (a) 3-edge junctions. (b) 2-edge junctions.

4 Affine Transformation Using JUDOCA

The next step is to establish correspondences between images. This objective is harder in case of wide baseline (or sparse view) pair of images. This is due to the existence of perspective deformation in such a case as mentioned above. Thus, a type of invariant measure should be used in order to achieve good results. The information inferred from the JUDOCA operator can be used to facilitate the process. This information includes the locations of junctions as well as the orientations of the edges forming them. Such an information may be utilized to approximate the accurate homographic transformation as a simpler affine version before performing cross correlation. (Other researchers tackled the affine invariance measure idea as in [14, 15].) Affine transformation matrix has 6 degrees of freedom (DOF's) and can be fully estimated using 3 pairs of points. In our case, every 2-edge junction forms a triangle (Fig. 4(b)) that is surrounded by 3 points; one point is the location of the junction and the other two points are the intersection between the two edges and the circular mask of JUDOCA. (Notice that a 3-edge junction can be split into three 2-edge junctions as in the example shown in Fig. 4(b).) Suppose that the points of the left junction are $\mathbf{m}_1 = [x_1, y_1]^T$, $\mathbf{m}_2 = [x_2, y_2]^T$, $\mathbf{m}_3 = [x_3, y_3]^T$ and of the right junction are $\mathbf{m}'_1 = [x'_1, y'_1]^T$, $\mathbf{m}'_2 = [x'_2, y'_2]^T$, $\mathbf{m}'_3 = [x'_3, y'_3]^T$ (see Fig. 5(a)).[†] The goal here is to find a 3×3 transformation matrix \mathbf{H} that maps a point in the source triangle to another in the destination triangle (as shown in Fig. 5(b)). Thus:

$$\mathbf{m}' = \mathbf{H}\mathbf{m} \quad (1)$$

where \mathbf{m} is a point in the left image and \mathbf{m}' is the corresponding right one. We can rewrite Eq. (1) as:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{h}_1^T \\ \mathbf{h}_2^T \\ \mathbf{h}_3^T \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (2)$$

*The JUDOCA operator is available at:

<http://www.site.uottawa.ca/school/research/viva/projects/judoca>

[†]The notation $\mathbf{m} = [x, y]^T$ represents an image point in inhomogeneous coordinates while $\mathbf{m} = [x, y, 1]^T$ represents the same point in homogeneous coordinates.

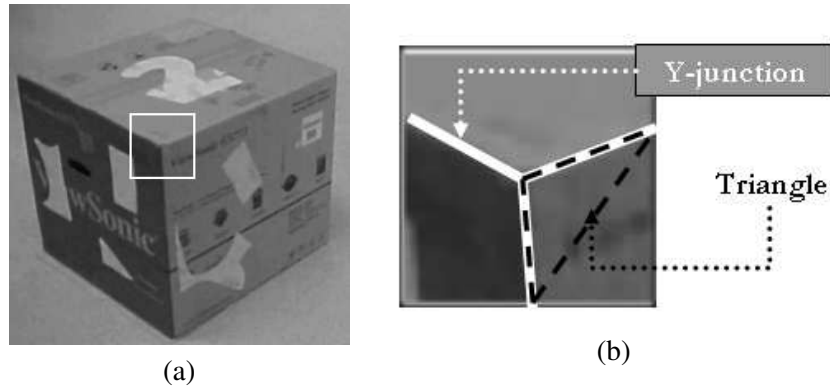


Figure 4: (a) One corner of a box that produces a Y-junction. (b) A Y-junction can be split into three 2-edge junctions. Each 2-edge junction form a triangle.

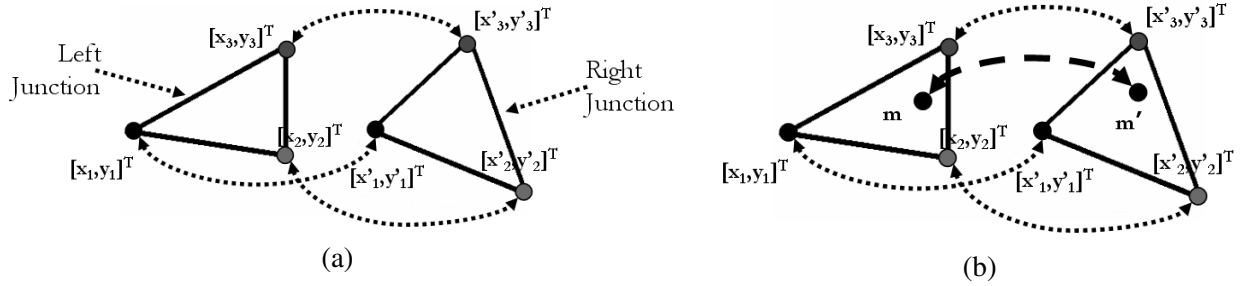


Figure 5: (a) Two corresponding junctions (or triangles) (b) Corresponding points within the triangles.

where $\mathbf{h}_1, \mathbf{h}_2$ and \mathbf{h}_3 represent the rows of the affine transformation matrix. As we have three pairs of points, we can write:

$$\begin{bmatrix} x'_1 \\ x'_2 \\ x'_3 \end{bmatrix} = \begin{bmatrix} \mathbf{m}_1^T \\ \mathbf{m}_2^T \\ \mathbf{m}_3^T \end{bmatrix} \mathbf{h}_1 = \begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{bmatrix} \begin{bmatrix} h_{11} \\ h_{12} \\ h_{13} \end{bmatrix} \tag{3}$$

Then, the first row in matrix H can be written as:

$$\mathbf{h}_1 = \begin{bmatrix} h_{11} \\ h_{12} \\ h_{13} \end{bmatrix} = \begin{bmatrix} \mathbf{m}_1^T \\ \mathbf{m}_2^T \\ \mathbf{m}_3^T \end{bmatrix}^{-1} \begin{bmatrix} x'_1 \\ x'_2 \\ x'_3 \end{bmatrix} = \begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{bmatrix}^{-1} \begin{bmatrix} x'_1 \\ x'_2 \\ x'_3 \end{bmatrix} \tag{4}$$

Similarly, the second and the third rows can be expressed as:

$$\mathbf{h}_2 = \begin{bmatrix} \mathbf{m}_1^T \\ \mathbf{m}_2^T \\ \mathbf{m}_3^T \end{bmatrix}^{-1} \begin{bmatrix} y'_1 \\ y'_2 \\ y'_3 \end{bmatrix} \tag{5}$$

$$\mathbf{h}_3 = \begin{bmatrix} \mathbf{m}_1^T \\ \mathbf{m}_2^T \\ \mathbf{m}_3^T \end{bmatrix}^{-1} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \tag{6}$$

Hence the affine transformation can be estimated to relate a point $\mathbf{m} = [x, y]$ to $\mathbf{m}' = [x', y']^T$ within the areas of the junctions as [4]:

$$\mathbf{m}' = \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \mathbf{H}\mathbf{m} = \begin{bmatrix} \begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{bmatrix}^{-1} \begin{bmatrix} x'_1 \\ x'_2 \\ x'_3 \end{bmatrix} \\ \begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{bmatrix}^{-1} \begin{bmatrix} y'_1 \\ y'_2 \\ y'_3 \end{bmatrix} \\ \begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{bmatrix}^{-1} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \end{bmatrix}^T \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} . \quad (7)$$

5 Affine Variance Normalized Correlation

Variance normalized correlation (VNC) provides reliable results over different viewing conditions [16] and can be used to cross-correlate different regions surrounding $[x, y]$ and $[x', y']$ in multiple views.

$$VNC_{(x,y;x',y')} = \frac{\sum_{m,n=-\frac{N}{2}}^{\frac{N}{2}} \mathcal{I}_l(x+m,y+n) - \overline{\mathcal{I}_l(x,y)} [\mathcal{I}_r(x'+m,y'+n) - \overline{\mathcal{I}_r(x',y')}]}{\sqrt{\sum_{m,n=-\frac{N}{2}}^{\frac{N}{2}} [\mathcal{I}_l(x+m,y+n) - \overline{\mathcal{I}_l(x,y)}]^2} \sqrt{\sum_{m,n=-\frac{N}{2}}^{\frac{N}{2}} [\mathcal{I}_r(x'+m,y'+n) - \overline{\mathcal{I}_r(x',y')}]^2}} \quad (8)$$

where $\overline{\mathcal{I}_l(x, y)}$ and $\overline{\mathcal{I}_r(x', y')}$ are the mean values of the neighborhoods surrounding $[x, y]^T$ and $[x', y']^T$ respectively. Notice that the previous equation assumes that correlation is performed on the image planes and inhomogeneous coordinates are used. Many researchers investigated this case; e.g., [5, 6]. In our case, there are two points to be taken into account.

1. Camera parameters are known within error margin limits. Thus, through the error margins of the sensor used, a strip surrounding the epipolar line in the right image may be established for each candidate in the left image. VNC correlation is applied among all candidates in the strip to determine the best match.
2. Correlation should not be applied to the image plane as in Eq. (8). Instead, that equation should be altered to deal with the perspective deformation existing. This is to be done by applying affine transformation prior to correlating the areas and performing the following check [3]:

$$\frac{\sum_{m,n=-\frac{N}{2}}^{\frac{N}{2}} [\mathcal{I}_l(x+m,y+n) - \overline{\mathcal{I}_l(x,y)}] [\mathcal{I}_r(\mathbf{k}^T) - \overline{\mathcal{I}_r(\mathbf{m}'^T)}]}{\sqrt{\sum_{m,n=-\frac{N}{2}}^{\frac{N}{2}} [\mathcal{I}_l(x+m,y+n) - \overline{\mathcal{I}_l(x,y)}]^2} \sqrt{\sum_{m,n=-\frac{N}{2}}^{\frac{N}{2}} [\mathcal{I}_r(\mathbf{k}^T) - \overline{\mathcal{I}_r(\mathbf{m}'^T)}]^2}} \geq t_{VNC} \quad (9)$$

where $N + 1$ is the side length of the correlation window in pixels; $\mathbf{m}' = \mathbf{H}\mathbf{m} = \mathbf{H}[x, y, 1]^T$; $\overline{\mathcal{I}_l(x, y)}$ is the mean of the neighborhood surrounding $[x, y]^T$; $\mathbf{k} = \mathbf{H}[x + m, y + n, 1]^T$; $\overline{\mathcal{I}_r(\mathbf{m}'^T)}$ is the mean of the neighborhood surrounding \mathbf{m}' ; and t_{VNC} is a threshold. Then, the best choice is the one that results in a true condition for the test (9) above with a maximum value among all candidates.

6 Deriving Rotation Matrix

So far, the previous procedure should produce a set of putative matches. Such a set may include outliers as well as good matches. At this point, it is required to convert the set of matches obtained into an accurate fundamental matrix. In order to achieve this, different methods can be implemented. Those methods can be classified into linear, iterative and robust methods [1]. Among them is the RANSAC scheme [7, 10] which is used for robust parameter estimation [19, 20]. RANSAC has the following steps.

1. Seven points are randomly selected.
2. The fundamental matrix is computed from this set where:

$$\mathbf{m}'^T \mathbf{F} \mathbf{m} = 0 \quad (10)$$

3. Pairs that satisfy the computed \mathbf{F} are selected to the support set. The pair is satisfying the matrix \mathbf{F} if the point in one image lies within some threshold, t_F , (e.g., 1 pixel) from the computed epipolar line. That is:

$$d(\mathbf{F} \mathbf{m}, \mathbf{m}') < t_F \quad (11)$$

where $d(\cdot)$ represents the distance from the point \mathbf{m}' to the epipolar line $l' = \mathbf{F} \mathbf{m}$; and t_F is a threshold.

If (11) results in a true condition, then the pair $(\mathbf{m}, \mathbf{m}')$ agrees with the matrix \mathbf{F} .

4. The preceding steps are repeated and the fundamental matrix, \mathbf{F} , that supports the largest match set is returned.

Notice that the accuracy of these steps depends in a way on the threshold t_F . The larger the threshold the larger the number of matching pairs to be accepted by RANSAC. This leads to the risk of inaccuracy. Thus, MLESAC [12] and MAPSAC [13] were proposed to avoid wrong results when using high thresholds with RANSAC. However, this is not the case in our proposal as in all our experiments, we do not use high thresholds.

Once the fundamental matrix has been estimated, the essential matrix \mathbf{E} can then be retrieved using the calibration matrix \mathbf{A} as:

$$\mathbf{E} = \mathbf{A}^T \mathbf{F} \mathbf{A} \quad (12)$$

Here we assumed that the calibration matrix is the same for the two images. Now, we aim to have the essential matrix decomposed such that:

$$\mathbf{E} = \mathbf{R} \mathbf{T}_\times \quad (13)$$

where \mathbf{R} is the accurate orthogonal rotation matrix and \mathbf{T}_\times is an anti-symmetric matrix defined as:

$$\mathbf{T}_\times = \begin{bmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & t_x & 0 \end{bmatrix} \quad (14)$$

where $\mathbf{T} = [t_x, t_y, t_z]$ represents the translation vector from one camera to the other. In order to get the decomposition of Eq. (13), the essential matrix can be decomposed using Singular Value Decomposition (SVD) to get:

$$\mathbf{E} = \mathbf{U} \mathbf{S} \mathbf{V}^T \quad (15)$$

The rotation matrix \mathbf{R} and the translation matrix \mathbf{T}_\times can then be expressed (up to a scale factor) as:

$$\begin{aligned} \mathbf{R} &= \mathbf{U} \mathbf{W} \mathbf{V}^T \text{ or } \mathbf{R} = \mathbf{U} \mathbf{W}^{-1} \mathbf{V}^T \\ \mathbf{T}_\times &= \mathbf{V} \mathbf{Z} \mathbf{V}^T \end{aligned} \quad \text{where } \mathbf{W} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \text{ and } \mathbf{Z} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (16)$$

The origin is assumed to be at the first camera and the rotation matrix R indicates the rotation from the first camera to the second one. The translation vector \dot{T} can be expressed as [8]:

$$\dot{T} = v \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad (17)$$

and

$$R\dot{T} = u \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad (18)$$

Note that the translation vector \dot{T} is normalized in this case. In addition, there are four possible rotation/translation pairs based on two possible choices of the rotation matrix and two possible signs of the translation vector. In other words, the projection matrix of the second camera may be one of the following [8]:

$$\begin{aligned} P' &= [UWV^T | U[0, 0, 1]^T] \\ P' &= [UWV^T | -U[0, 0, 1]^T] \\ P' &= [UW^T V^T | U[0, 0, 1]^T] \\ P' &= [UW^T V^T | -U[0, 0, 1]^T] \end{aligned} \quad (19)$$

7 Rotation Angles in Different Systems

The previous step should result in an accurate 3×3 rotation matrix. Such a matrix expresses the rotation from the first camera to the second one. Rotation angles are embedded in this matrix and can be extracted from it. There are different rotation systems that can be used to express the rotation between cameras. These systems share some properties:

1. Any system should result in the same rotation matrix.
2. A system requires the existence of three rotation angles.
3. A 3D rotation can be expressed as a series of three 2D rotations.

In this paper, we will consider two systems; using ω, ϕ and κ angles as compared to tilt, τ , pan, ρ , and swing, ψ . Some systems (including commercial) are using only one system to express the rotation angles; thus, we are including both systems to show the easy conversion between them.

Considering the ω, ϕ and κ angles, the 3D rotation is performed in three 2D rotations. Fig. 6(a) shows the first rotation through an angle ω about the X -axis. The second rotation through an angle ϕ about the rotated Y -axis is depicted in Fig. 6(b). Finally, the third rotation is through an angle κ about the rotated Z -axis as shown in Fig. 6(c). The final rotation is calculated by concatenating the three rotation matrices. That is:

$$R = R_{\kappa} R_{\phi} R_{\omega} = \begin{bmatrix} \cos \phi \cos \kappa & \sin \omega \sin \phi \cos \kappa + \cos \omega \sin \kappa & -\cos \omega \sin \phi \cos \kappa + \sin \omega \sin \kappa \\ -\cos \phi \sin \kappa & -\sin \omega \sin \phi \sin \kappa + \cos \omega \cos \kappa & \cos \omega \sin \phi \sin \kappa + \sin \omega \cos \kappa \\ \sin \phi & -\sin \omega \cos \phi & \cos \omega \cos \phi \end{bmatrix} \quad (20)$$

Fig. 7 shows another system of angles; pan, ρ , tilt, τ , and swing, ψ [18]. As in the previous case, the overall 3D rotation is performed in three separate sequential 2D rotations about the Z -axis, the X -axis and then the Z -axis again. The final rotation matrix is calculated as:

$$R = R_{\psi} R_{\tau} R_{\rho} = \begin{bmatrix} \cos \psi \cos \rho + \sin \psi \cos \tau \sin \rho & -\cos \psi \sin \rho + \sin \psi \cos \tau \cos \rho & \sin \psi \sin \tau \\ -\sin \psi \cos \rho + \cos \psi \cos \tau \sin \rho & \sin \psi \sin \rho + \cos \psi \cos \tau \cos \rho & \cos \psi \sin \tau \\ -\sin \tau \sin \rho & -\sin \tau \cos \rho & \cos \tau \end{bmatrix} \quad (21)$$

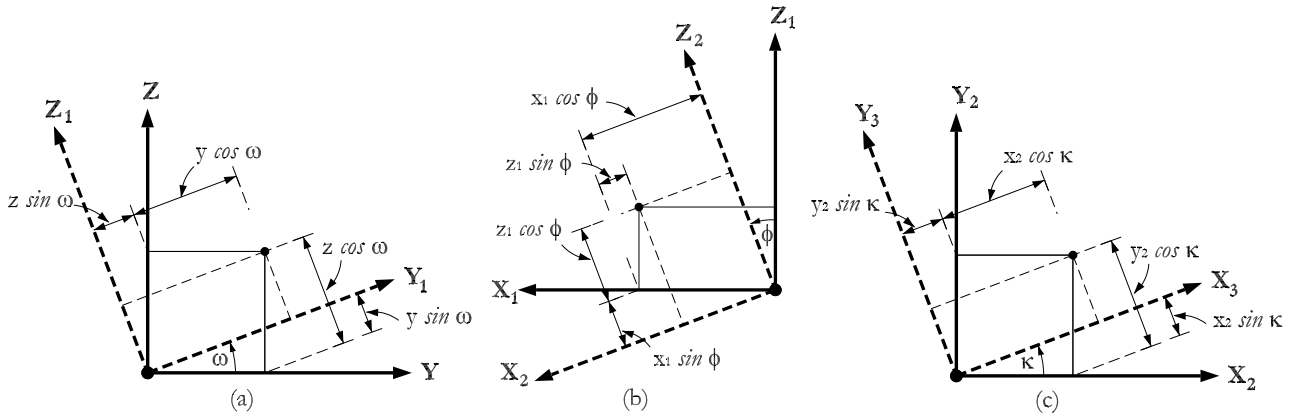


Figure 6: (a) Rotation through R_ω . (b) Rotation through R_ϕ . (c) Rotation through R_κ .

Both Eqs. (20) and (21) should result in the same rotation matrix. Thus, using those equations, the values of the tilt, τ , the pan, ρ and the swing, ψ angles given ω , ϕ and κ can be calculated [18].

$$\begin{aligned}
 \tau &= \cos^{-1}(r_{33}) = \cos^{-1}(\cos \omega \cos \phi) \\
 \rho &= \tan^{-1}\left(\frac{-r_{31}}{-r_{32}}\right) = \tan^{-1}\left(\frac{-\sin \phi}{\sin \omega \cos \phi}\right) \\
 \psi &= \tan^{-1}\left(\frac{r_{13}}{r_{23}}\right) = \tan^{-1}\left(\frac{-\cos \omega \sin \phi \cos \kappa + \sin \omega \sin \kappa}{\cos \omega \sin \phi \sin \kappa + \sin \omega \cos \kappa}\right)
 \end{aligned} \tag{22}$$

The same idea can be used to calculate ω , ϕ and κ given τ , ρ and ψ .

$$\begin{aligned}
 \omega &= \tan^{-1}\left(\frac{-r_{32}}{r_{33}}\right) = \tan^{-1}\left(\frac{\sin \tau \cos \rho}{\cos \tau}\right) \\
 \phi &= \sin^{-1}(r_{31}) = \sin^{-1}(-\sin \tau \sin \rho) \\
 \kappa &= \tan^{-1}\left(\frac{-r_{21}}{r_{11}}\right) = \tan^{-1}\left(\frac{\sin \psi \cos \rho - \cos \psi \cos \tau \sin \rho}{\cos \psi \cos \rho + \sin \psi \cos \tau \sin \rho}\right)
 \end{aligned} \tag{23}$$

Note that the previous method of enhancing the rotation matrix results in measurements with respect to the first camera. In this setup, the origin of the coordinate system is at the optical center of the first camera, $\dot{\mathbf{C}}$, and the rotation angles are measured accordingly. Furthermore, the translation vector obtained (i.e., the location of the second camera, $\dot{\mathbf{C}}'$) is measured from that origin and normalized.

8 Experimental Results

In this section, we will show how our procedure can be effective in enhancing the parameters estimated. In order to show this quantitatively, we compare the results produced through camera parameters estimated using PhotoModeler and the results after enhancement. PhotoModeler is used to get the camera parameters for the stereo pairs shown in Fig. 8(a) and (b) and Fig. 9(a) and (b). The parameters obtained are listed in Table 1 and Table 2.

The fundamental matrix can, then, be calculated through camera parameters as [2]:

$$\begin{aligned}
 \mathbf{F} &= [\mathbf{A}'\mathbf{R}']^{-1T}[\dot{\mathbf{C}} - \dot{\mathbf{C}}']_{\times}[\mathbf{A}\mathbf{R}]^{-1} \\
 &= \left[[\mathbf{A}'\mathbf{R}'][\dot{\mathbf{C}} - \dot{\mathbf{C}}'] \right]_{\times} [\mathbf{A}'\mathbf{R}']^{-1} \\
 &= [\mathbf{e}']_{\times} \mathbf{P}'\mathbf{P}^+
 \end{aligned} \tag{24}$$

where \mathbf{A} and \mathbf{R} are the calibration and rotation matrices of the left camera respectively; and \mathbf{A}' and \mathbf{R}' are their counterparts of the right camera; $\dot{\mathbf{C}}$ and $\dot{\mathbf{C}}'$ are the optical centers; and \mathbf{e}' is the right epipole.

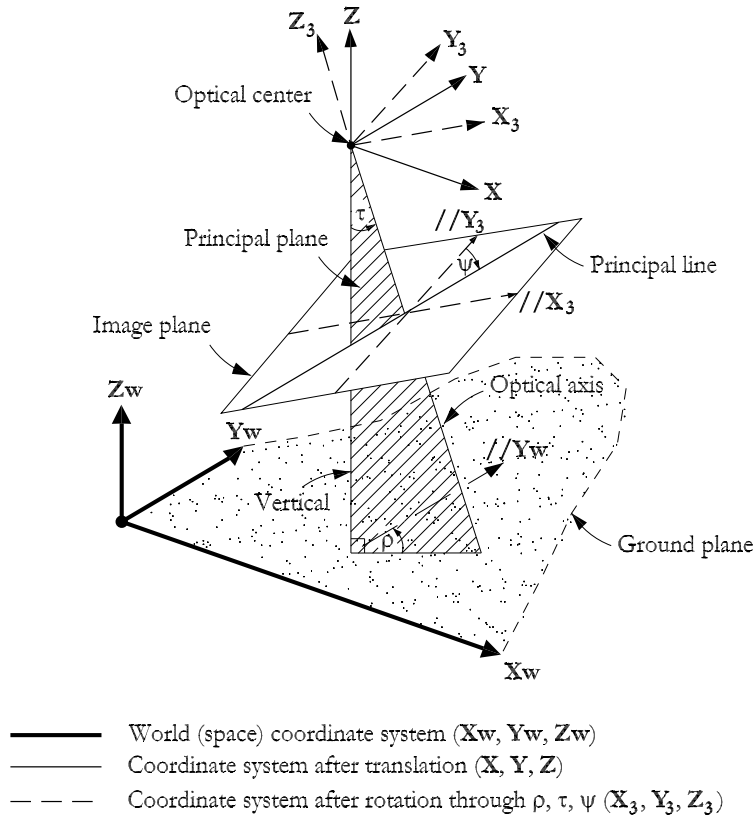


Figure 7: Pan, ρ , tilt, τ , and swing, ψ , rotation angles. The angle – in the principal plane – between the optical axis and the vertical line is called the tilt, τ . The angle – in the ground plane – between the direction of the Y_w -axis and intersection between the principal plane and the ground plane is called the pan, ρ (also known as *azimuth*). The angle – in the image plane – between the direction of Y_3 -axis and the principal line is called the swing, ψ .

Also, the fundamental matrix can be expressed as [3]:

$$F = [e']_{\times} [A'R'] [AR]^{-1} \tag{25}$$

In our experiments, we use all the formulae of Eq. (24) and Eq. (25) with camera parameters produced by PhotoModeler. All resulted in the same fundamental matrix. Comparisons are performed along two tracks. The first track is by showing the differences between the fundamental matrices calculated through parameters produced by PhotoModeler and the fundamental matrices tuned after the correlation method. The second track we use is by calculating the differences between rays passing through corresponding points and should theoretically intersect in space while they do not because of the inaccurate camera parameters.

8.1 F Differences

The measure of comparison that we use was proposed by Stephane Laveau from INRIA Sophia-Antipolis and detailed in [20]. It was meant to characterize the difference between two fundamental matrices. The overall idea is to measure the differences between points and corresponding epipolar lines on image planes. Suppose that the two fundamental matrices are F_1 and F_2 , then considering Fig. 10, the steps are the following:

1. A point $\mathbf{m} = [x, y, 1]^T$ is chosen randomly in the left image.
2. The epipolar line $F_1 \mathbf{m} = [a_1, b_1, c_1]^T$ can be drawn in the right image as shown in Fig 10.

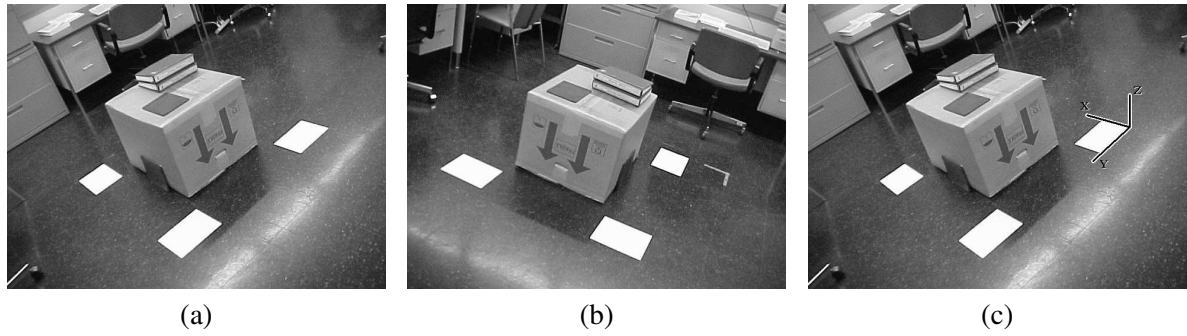


Figure 8: (a) and (b) The “box” pair. (c) The world coordinate system is shown where the world origin is located at a corner of a paper on the ground surface and where the XY -plane coincides with the ground surface and the Z -axis is vertical.

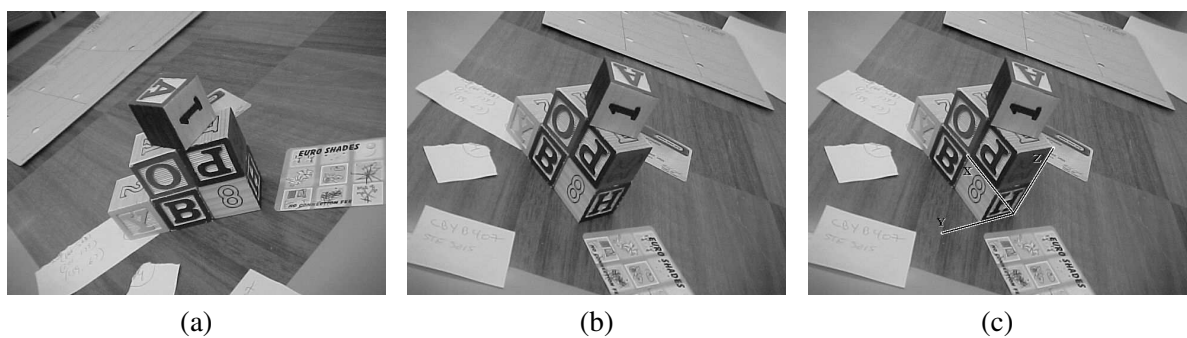


Figure 9: (a) and (b) The “cubes” pair. (c) The world origin is shown at the corner of a bottom cube where the XY -plane coincides with the ground surface and the Z -axis is vertical.

Image	X	Y	Z	ω	ϕ	κ
Fig. 8(a)	-1535.2119	1720.8447	1693.2169	-35.1616	-46.7723	-155.5747
Fig. 8(b)	-1694.5607	-142.2461	1585.1900	25.0285	-55.5578	-47.0806

Table 1: The “box” pair: The parameters for each camera estimated by PhotoModeler. The world origin is located as shown in Fig. 8(c) where the Z -axis is vertical or perpendicular to the ground plane. The units are in millimeters for X , Y , Z and in degrees for ω , ϕ and κ . Each image is 640×480 pixels. The focal length is 4.3 mm. The format size is 3.5311×2.7446 mm.

Image	X	Y	Z	ω	ϕ	κ
Fig. 9(a)	-114.7469	398.9921	355.3564	-52.2783	-17.0877	162.6817
Fig. 9(b)	-206.6808	372.5378	351.6793	-49.5184	-27.0316	-132.0044

Table 2: The “cubes” pair: The parameters for each camera estimated by PhotoModeler. The world origin is located as shown in Fig. 9(c). The Z -axis is vertical or perpendicular to the ground plane. The units are in millimeters for X , Y , Z and in degrees for ω , ϕ and κ . Each image is 640×480 pixels. The focal length is 6.0 mm. The format size is 3.5311×2.7446 mm.

3. A check should be made to ensure that the line $F_1 \mathbf{m}$ intersects the right image. If this is not the case, start over.
4. A point $\mathbf{m}' = [x', y', 1]^T$ is selected randomly on $F_1 \mathbf{m}$.
5. The epipolar line $F_2 \mathbf{m} = [a_2, b_2, c_2]^T$ is to be drawn in the right image using F_2 and the distance d'_1 is calculated as [17]:

$$d'_1 = \frac{|a_2 x' + b_2 y' + c_2|}{\sqrt{a_2^2 + b_2^2}} \quad (26)$$

6. An epipolar line that corresponds to \mathbf{m}' is drawn in the left image. This epipolar is estimated as $F_2^T \mathbf{m}'$. The perpendicular distance d_1 can be easily estimated as done before.
7. Steps 2 through 6 are repeated while reversing the roles of F_1 and F_2 and the distances d_2 and d'_2 are calculated as done with d_1 and d'_1 .
8. All the previous steps are repeated N times.
9. The difference between the two fundamental matrices is estimated as the average of all d 's.

The above procedure was applied to the “box” stereo pair shown in Fig. 11(a) and Fig. 11(b). The N used was 1000. The results of the differences are summarized in Table 3. Another stereo pair with its results are shown in Fig. 12.

8.2 Ray Differences

At this point, we know that there is a difference between a fundamental matrix calculated through parameters estimated using PhotoModeler and another fundamental matrix estimated after enhancement; however, which is more accurate?

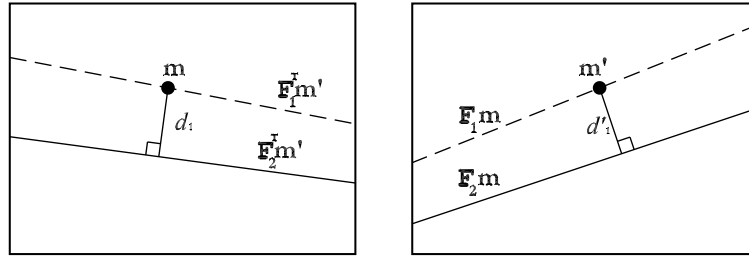


Figure 10: Estimating difference between fundamental matrices on image planes.

#points	d_1	d'_1	d_2	d'_2	Avg
100	6.98	7.917	7.256	7.799	7.488
200	6.722	7.516	6.921	7.403	7.141
300	6.424	7.198	6.768	7.329	6.93
400	6.425	7.109	6.861	7.393	6.947
500	6.413	7.134	6.78	7.344	6.918
600	6.384	7.109	6.691	7.261	6.861
700	6.444	7.197	6.678	7.248	6.892
800	6.509	7.27	6.738	7.332	6.962
900	6.451	7.184	6.733	7.328	6.924
1000	6.498	7.251	6.694	7.282	6.931

Table 3: Differences between fundamental matrices for the pair shown in Fig. 8(a) and Fig. 8(b).

We may use triangulation to assess how accurate our measurements are. The optimal triangulation case is depicted in Fig. 13(a). In such a case, the rays passing through the optical centers and the corresponding points in images intersect in space at the location of the 3D point. In most cases and due to different reasons like inaccuracy in measurements or existence of image noise, the rays are likely to be skew as depicted in Fig. 13(b). If there is no severe inaccuracy, the perpendicular to both rays will be short; otherwise, the greater the inaccuracy in measurements the farther the rays will be from each other. Thus, in case of ray skewness, the shortest distance between the skew rays (δ in Fig. 13(b)) should give an indication of measurement accuracy. Short distances that tend to zero declare good accuracy.

In order to estimate the shortest distance between two skew rays, the vector equations of those rays can be expressed as:

$$\dot{\mathbf{R}} = \dot{\mathbf{C}} + \lambda \dot{\mathbf{D}} \text{ and } \dot{\mathbf{R}}' = \dot{\mathbf{C}}' + \lambda' \dot{\mathbf{D}}' \quad (27)$$

where $\dot{\mathbf{C}}$ and $\dot{\mathbf{C}}'$ are the optical centers; λ and λ' are scaling factors; and $\dot{\mathbf{D}}$ and $\dot{\mathbf{D}}'$ and are the direction vectors which can be calculated as [2]:

$$\dot{\mathbf{D}} = \mathbf{A}^{-1} \mathbf{m} \text{ and } \dot{\mathbf{D}}' = (\mathbf{A}\mathbf{R})^{-1} \mathbf{m}' \quad (28)$$

where \mathbf{A} is the calibration matrix; \mathbf{R} is the rotation matrix we estimated; and \mathbf{m} and \mathbf{m}' are a pair of corresponding points. Finally, the shortest distance δ between the rays if they are skew can be expressed as:

$$\delta = \frac{\dot{\mathbf{D}} \times \dot{\mathbf{D}}'}{\|\dot{\mathbf{D}} \times \dot{\mathbf{D}}'\|} \bullet \dot{\mathbf{C}}' \quad (29)$$

where $\|\cdot\|$ indicates the norm; and \times and \bullet are the cross and dot products respectively.

We tested our algorithm on groups of correct matches obtained for the pairs shown in Fig. 8(a) and (b) and in Fig. 9(a) and (b). As a comparison, we tested the differences between skew rays through the parameters acquired using PhotoModeler as opposite to the differences obtained after enhancement of angles. The results are shown as bar charts in Fig. 14(a) for the “box” pair and in Fig. 14(b) for the “cubes” pair. The differences

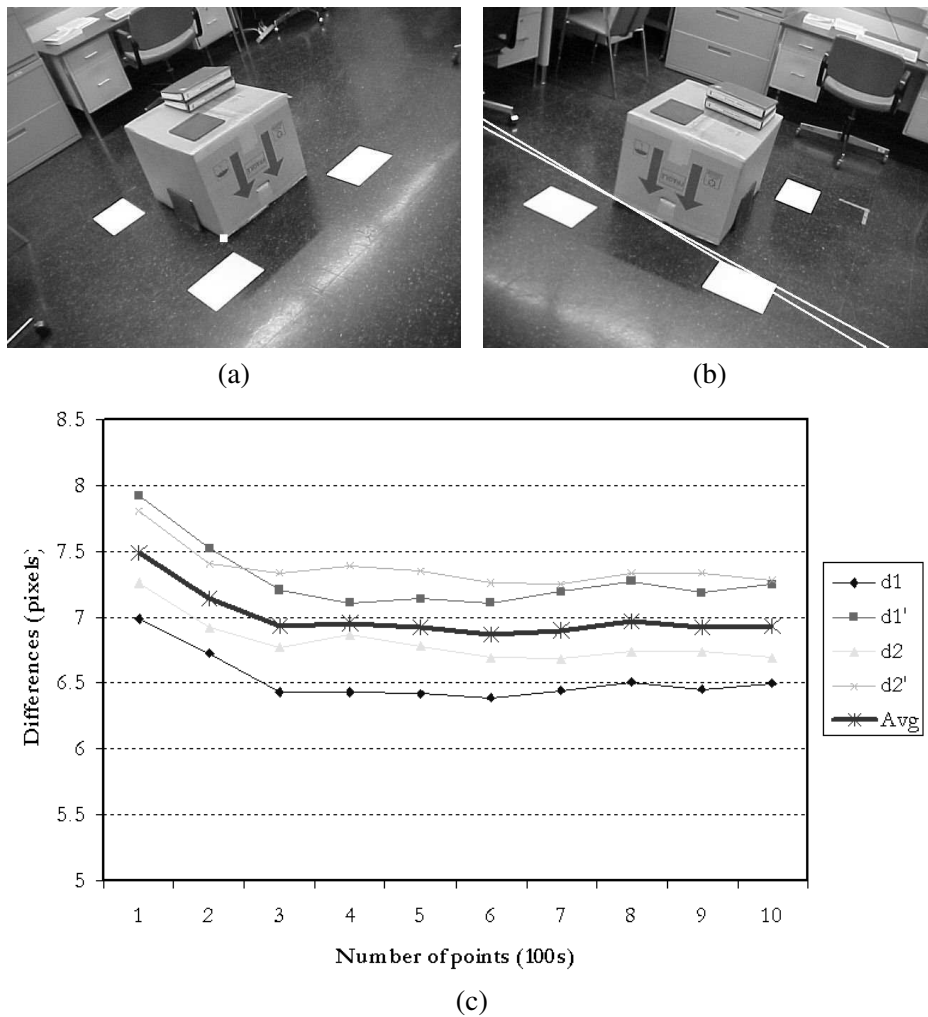


Figure 11: The “box” pair. (a) Left image with a point marked. (b) Right image with epipolar lines obtained through parameters estimated using PhotoModeler and after enhancement. (c) Differences on image planes between fundamental matrices calculated using PhotoModeler parameters and our proposed method.

are noticeable in both cases for individual corresponding pair of points. The overall differences are shown in Fig. 14(c), which clearly demonstrates the effectiveness of enhancement.

9 Conclusions

Accuracy of camera parameters including rotation angles affects many Computer Vision tasks. In this paper, we suggested a sequence of steps to enhance the accuracy of camera rotation angles given a pair of wide baseline images. We utilized information inferred by the JUDOCA junction detector in order to perform a cross-correlation matching process that overcame the difficulty of the wide baseline case under consideration. We used the obtained match set to estimate the fundamental and essential matrices. The later matrix was decomposed into a rotation and a translation matrices. Details of expressing the angles calculated from the rotation matrix were explained in two different rotation systems. Finally, test measures were applied and the overall accuracy was compared to that of PhotoModeler.

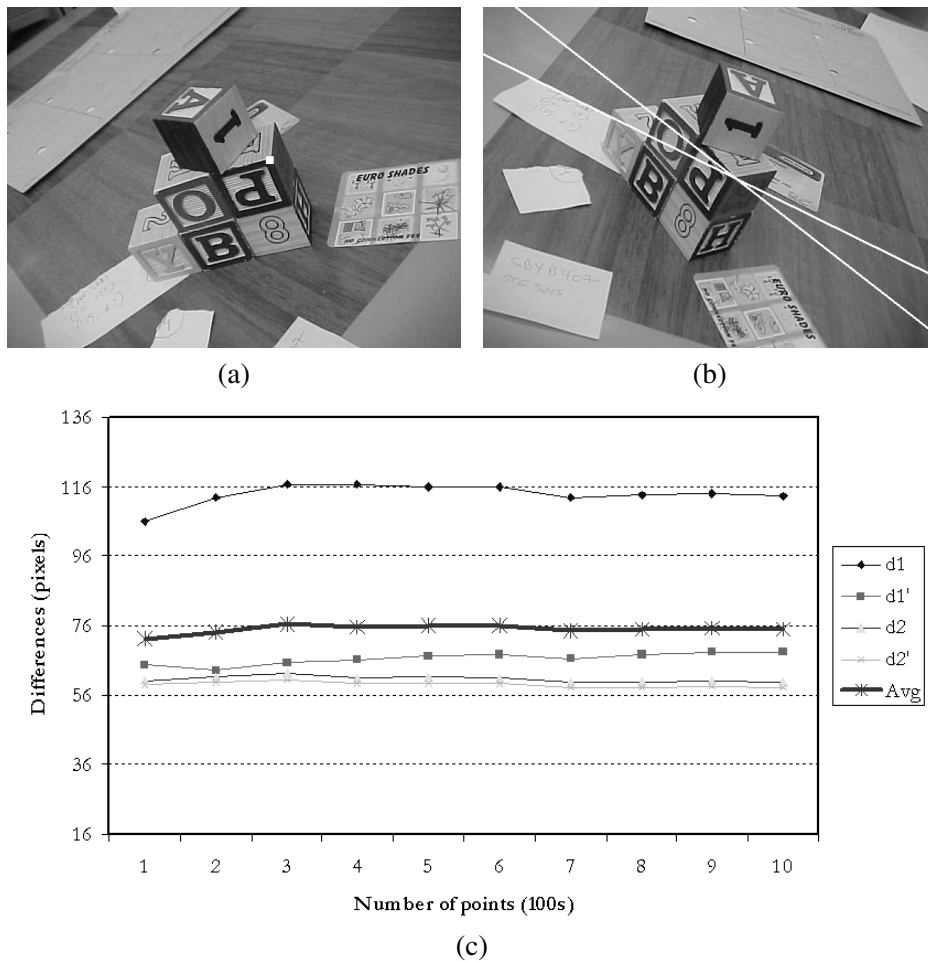


Figure 12: The “cubes” pair. (a) Left image with a point marked. (b) Right image with epipolar lines obtained through parameters estimated using PhotoModeler and after enhancement. (c) Differences on image planes between fundamental matrices calculated using PhotoModeler parameters and our proposed method. This pair of images was shot by Prof. R. Laganieri of the University of Ottawa, Canada.

References

- [1] X. Armangue and J. Salvi. Overall view regarding fundamental matrix estimation. *Image and Vision Computing*, 21(2):205–220, February 2003.
- [2] P. A. Beardsley, A. Zisserman, and D. W. Murray. Sequential Updating of Projective and Affine Structure from Motion. *International Journal of Computer Vision*, 23(3):235–259, 1997.
- [3] R. Elias. *Towards Obstacle Reconstruction Through Wide Baseline Set of Images*. Ph.D. Thesis, University of Ottawa, Canada, 2004.
- [4] R. Elias and A. Elnahas. An accurate indoor localization technique using image matching. In *Proceedings the 3rd IET International Conference on Intelligent Environments, IE'07*, Ulm, Germany, September 2007.

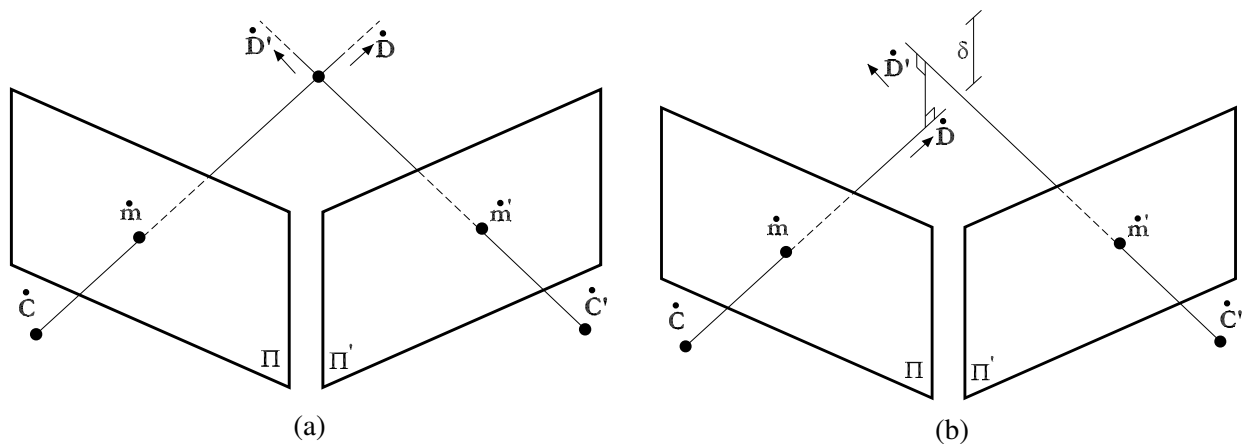
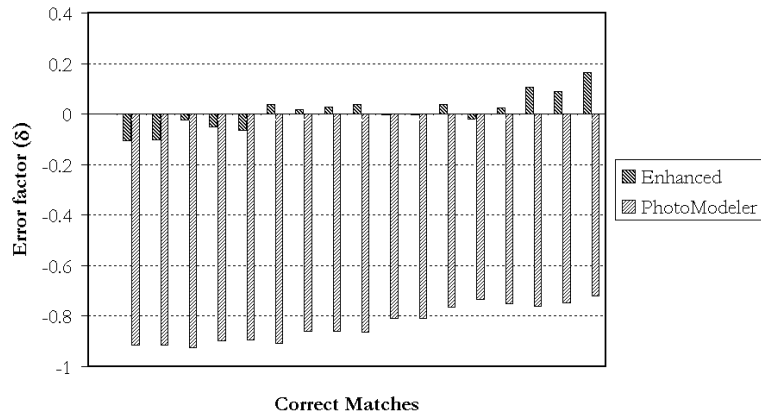


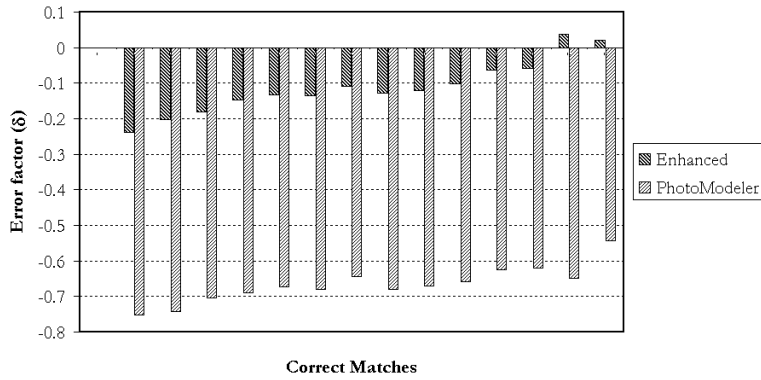
Figure 13: Triangulation. (a) Optimal case. (b) Regular case.

- [5] L. Falkenhagen. Depth estimation from stereoscopic image pairs assuming piecewise continuous surface. In *Image Processing for Broadcast and Video Production*, pages 115–127, 1994.
- [6] O. Faugeras. *Three-Dimensional Computer Vision, A Geometric Viewpoint*. The MIT Press, Cambridge, Massachusetts, 1996.
- [7] M. A. Fischler and R. C. Bolles. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Comm. of the ACM*, 24:381–395, 1981.
- [8] Richard I. Hartley. Estimation of relative camera positions for uncalibrated cameras. In *ECCV '92: Proceedings of the Second European Conference on Computer Vision*, pages 579–587, London, UK, 1992. Springer-Verlag.
- [9] R. Laganière and R. Elias. The Detection of Junction Features in Images. In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP'04*, volume III, pages 573–576, 2004.
- [10] G. Roth and A. Whitehead. Using Projective Vision to Find Camera Positions in an Image Sequence. In *Proceedings Vision Interface 2000*, pages 255–232, 2000.
- [11] K. Stum. Sensor accuracy and calibration theory and practical application. In *Proceedings the 14th National Conference on Building Commissioning*, San Francisco, California, April 2006.
- [12] P. Torr and A. Zisserman. Mlesac: A new robust estimator with application to estimating image geometry. *Computer Vision and Image Understanding*, 78(1):138–156, April 2000.
- [13] P.H.S. Torr. Bayesian model estimation and selection for epipolar geometry and generic manifold fitting. *International Journal of Computer Vision*, 50(1):35–61, October 2002.
- [14] T. Tuytelaars and L. Van Gool. Matching Widely Separated Views based on Affinely Invariant Neighbourhoods. *Submitted to International Journal on Computer Vision*, July 2001.
- [15] T. Tuytelaars, L. Van Gool, L. D'haene, and R. Koch. Matching of Affinely Invariant Regions for Visual Servoing. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1601–1606, 1999.

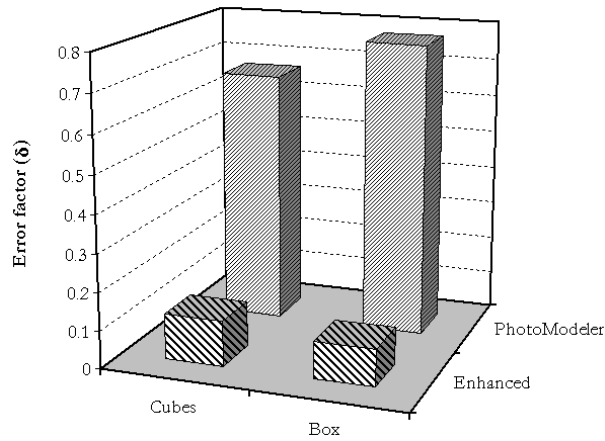
- [16] E. Vincent and R. Laganière. Matching feature points in stereo pairs: A comparative study of some matching strategies. *Machine Graphics and Vision*, 10(3):237–259, 2001.
- [17] E. Weisstein. Point-line distance–2-dimensional. MathWorld–A Wolfram Web Resource.
- [18] P. Wolf and B. Dewitt. *Elements of Photogrammetry*. McGraw Hill, 2000.
- [19] Z. Zhang. Determining the epipolar geometry and its uncertainty: A review. Technical Report 2927, Sophia-Antipolis Cedex, France, 1996.
- [20] Z. Zhang. Determining the epipolar geometry and its uncertainty: A review. *International Journal of Computer Vision*, 27(2):161–198, 1998.



(a)



(b)



(c)

Figure 14: Error factor for a group of correct matches. (a) The “box” pair. (b) The “cubes” pair. (c) Overall average.