# Accurate Image Depth Determination for Autonomous Vehicle Navigation

**Jurek Z. Sasiadek and Mark J. Walker**[1]

[1] Department of Mechanical and Aerospace Engineering
Carleton University
Ottawa, Ontario, Canada

mark_walker@carleton.ca

**Abstract**
This paper examines accurate image depth determination in order to achieve accurate navigation. This is critical for Unmanned Aerial Vehicle or UAV aerial refueling and also space debris clearance operations. The theory behind image depth calculation is explained and then synthetic pixel data is manufactured in order to determine a 95% confidence interval on image depth under various errors conditions. It is shown that without image rectification accurate depth can only be calculated for stereo cameras pointing without any axial rotations. With image rectification and under the stated conditions, image depth can be calculated with 6% or less relative error. Future research is discussed.

## 1. Introduction

The authors' research interests encompass the accurate calculation of pose – position and attitude – of an Unmanned Aerial Vehicle (UAV) in an urban environment. The accurate determination of this pose 6-tuple (x, y, z, pitch, roll, yaw) is important for UAV navigation. More particularly, for example, accurate navigation is crucial to autonomous aerial refueling. These problems have recently received attention [1]-[3]. It should also be noted that accurate navigation is vital to space-based robotic vehicles used for space debris clearance operations. It has been discussed [4]-[5] that the Global Positioning System (GPS) is the most widely used sensor for aircraft navigation. These papers have pointed out that GPS signals can be blocked. As well, the Volpe Report [6] has also reported that GPS signals can be blocked intentionally and unintentionally, by the leaves of a tree, for example, or a building. It is reasonable to enquire whether other sensors might be used in place of, or as an enhancement to GPS.

The pose of a UAV can be estimated using an Inertial Navigation Unit (INU). However, linear position is derived through a double integration of the linear acceleration which is an output of these devices. Consequently, small errors in acceleration quickly result in large cumulative position errors. In addition, if the vehicle undergoes small accelerations, then small errors will swamp the small accelerations, leading to large errors in position. Attitude (pitch, roll, yaw) can be

reliably obtained from an INU and altitude (z) can be derived from pressure sensors. Thus, position in the (x,y) plane still needs to be estimated. It is common for UAVs and mini- and micro-UAVs (MAVs) to be equipped with cameras. Can these be used to estimate position in the xy-plane?

UAV navigation through machine vision has been studied in [2]-[5] and [7]-[10]. It is assumed that two images of the environment through which the UAV is flying have been taken. A projective transformation called a homography is defined that maps the pixels of the first-in-time image into the pixels of the second-in-time image. Rotation and translation can be extracted from such a homography. Deriving this homography is a three step process. In the first, points-of-interest or features are extracted from each image [11]. In the second step, corresponding points-of-interest or features are found across the two images [12]. Finally, a homography is defined from the corresponding points-of-interest [12]. Rotation and translation can be derived if image depth is known [13]-[14. The process of estimating depth has been discussed [15]-[16]. A more robust technique is to rectify the images [17]-[18] first and then to estimate depth [17].

It is clear that in order to estimate rotation and translation it is necessary to find many, at least 4, matched points-of-interest across the two consecutive images [12]. Unmanned Aerial Vehicle urban travel is important since it is rich in corners that may be tracked by an on-board vision system.

In this research synthetic pixel data is created for which the depth is known to a high degree of accuracy. The goal of the research is to report under highly controlled conditions the exact (within a statistical confidence interval) error achieved with software that implements the theory explained below.

The remainder of this paper is organized as described. Section II explains the theory of depth calculation using a stereo camera. Section III explains how test pixels are generated. Section IV presents the results and section V describes the conclusions.

## 2. Estimating Feature Depth

The authors have published several papers [19] – [23] on the problem of accurately finding corners in images and then matching these across images-in-time. Depth of an image point and camera pose is dependent upon finding these matches. These problems will not be considered further in this paper.

Assuming two consecutive-in-time images I and I', with feature pixels $x_i = [u_i\ v_i\ 1]^T$ from image I and their matching feature pixels $x_i' = [u_i'\ v_i'\ 1]^T$ from image I', depth calculation proceeds in three steps. First the fundamental matrix is estimated between I and I', then the two images are rectified. Finally, the depth may be estimated.

## 2.1. Estimating the Fundamental Matrix

The fundamental matrix, F, between images I and I' satisfies:

$$x_i'^{T} F x_i = 0 \tag{2.1-1}$$

for any pair of matched pixels, $x_i$ and $x_i'$. When there are at least 8 of these, F can be estimated as described in Chapter 11 [12]. In particular when (2.1-1) is expanded:

$$Af = 0$$

$$\begin{bmatrix} x_1'x_1 & x_1'y_1 & x_1' & y_1'x_1 & y_1'y_1 & y_1' & x_1 & y_1 & 1 \\ . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . \\ x_n'x_n & x_n'y_n & x_n' & y_n'x_n & y_n'y_n & y_n' & x_n & y_n & 1 \end{bmatrix} f = 0 \tag{2.1-2}$$

$$f = [f_{11} \quad f_{12} \quad f_{13} \quad f_{21} \quad f_{22} \quad f_{23} \quad f_{31} \quad f_{32} \quad f_{33}]$$

for $n \geq 8$ pixel matches across images I and I'.

In an ideal world, (2.1-1) is obeyed by every corresponding match and a solution to (2.1-2) may be exactly found. In general, of course, each pixel in both images, will have an error and (2.1-2) can only be found as the solution minimizing some cost function.

The singular value decomposition can be used to solve (2.1-2) as the solution minimizing the least squares error, for which $\|f\| = 1$ [12].The fundamental matrix is singular, while the SVD solution to (2.1-2) is not necessarily [12]. If F is the SVD solution, it is replaced by the matrix F' for which det F' = 0 and that minimizes the Frobenius norm $\|F-F'\|$ [12]. While this method is simple and easy to implement, it is not numerically optimal because it treats all the entries of F as of equal importance [12]. Thus an algebraic minimization algorithm is used to find the singular matrix F' that minimizes $\|Af'\|$ for which $\|f'\| = 1$.

The algebraic minimization algorithm requires an initial estimate of F' [12]. This is arrived at with the normalized 8-point algorithm [12]. Normalization of $n \geq 1$ pixels $[u_i \; v_i \; 1]^T$ in an image is achieved by applying a transformation with the characteristics listed below, to all the pixels of that image [12].

- Translate all pixels so that their centroid or arithmetic mean is at the origin;
- Scale all pixels so that the root mean square (RMS) error is $\sqrt{2}$.

The normalized 8-point algorithm is algorithm 11.1 of [12]. The algebraic minimization is solved using algorithm 11.2 [12].

## 2.2. Rectifying Images

With an accurate estimate of F, rectifying homographies may now be calculated for the first-in-time image, $H_L$, and the second-in-time image, $H_R$ as described in chapter 11 [17].

## 2.3. Estimating Image Depth

The two rectifying homographies are applied to the. pixels $x_i = [u_i\ v_i\ 1]^T$ from image I and their matching feature pixels $x_i' = [u_i'\ v_i'\ 1]^T$

$$\begin{aligned} \hat{x}_i &= H_L x_i \\ \hat{x}_i' &= H_R x_i' \end{aligned} \tag{2.3-1}$$

and the fundamental matrix $\hat{F}$ is estimated. Note that if rectification is not to be employed then $H_L$ and $H_R$ are replaced by the Identity matrix. These rectified pixels may contain errors; they can be corrected as described in algorithm 12.1 [12].
The camera matrices, $C_L$ and $C_R$, are now obtained as described in chapter 11 [17]. The rotation and translation of the left camera in camera coordinates is assumed known, $R_L$ and $T_L$, as well as the right, $R_R$ and $T_R$. As well, the camera calibration matrix, K, as assumed known and is applicable to the left and right cameras, then

$$\begin{aligned} C_L &= H_L K R_L \begin{bmatrix} I & 0 \end{bmatrix} \\ C_R &= H_R K R_R \begin{bmatrix} I & -(T_R - T_L) \end{bmatrix} \end{aligned} \tag{2.3-2}$$

Triangulation is now carried out to calculate the 3D point for each matched and rectified pair of feature pixels, as described in chapter 12 [12]. For a given pair of matched and rectified pixels and assuming the subscript j denotes the j'th row of $C_L$ and $C_R$ and assuming $X = [x\ y\ z\ w]^T$ is the 3D point in homogeneous coordinates, then

$$\begin{aligned} AX &= 0 \\ A &= \begin{bmatrix} \hat{u}_i C_L^3 - C_L^1 \\ \hat{v}_i C_L^3 - C_L^2 \\ \hat{u}_i' C_R^3 - C_R^1 \\ \hat{v}_i C_R^3 - C_R^2 \end{bmatrix} \end{aligned} \tag{2.3-3}$$

A solution for X can be found using SVD and depth returned as X[3] / X[4.

## 2.4. Estimating the Camera Calibration Matrix

The camera calibration matrix, K, describes the transformation of the location of a 3D scene point into a 2D image point under a particular imaging geometry, unique for a given camera and its settings. Camera calibration produces intrinsic and extrinsic parameters [24] and has received much attention in the literature [25]-[28]. In particular the toolbox of J. Bouguet at Caltech [24] has had multiple releases and it was decided to use this toolbox. The intrinsic camera parameters produced by this toolbox are listed below.

- Focal length in pixels, stored in 2x1 vector **fc**
- Principal point in pixels, stored in 2x1 vector **cc**
- Skew coefficient, stored in scalar **alpha_c**
- Image radial and tangential distortion coefficients, stored in the 5x1 vector **kc**

Using the conventions of [24], the camera matrix is defined as:

$$K = \begin{bmatrix} fc(1) & alpha\_c * fc(1) & cc(1) \\ 0 & fc(2) & cc(2) \\ 0 & 0 & 1 \end{bmatrix} \tag{2.4-1}$$

# 3. Test Pixels

Fig.3-1 depicts the two co-ordinate systems relevant to the reported research. In the material that follows superscripts denote the co-ordinate system the value is relative to – E for Earth, C for camera – while subscripts denote the quantity, as explained. It is assumed that the left and right cameras, $C_L$ and $C_R$ respectively, have the same camera matrix, K. The rotation matrices $R_{EC}$ and $R_{CE}$ rotate earth-to-camera and camera-to-earth co-ordinates, respectively.

If the camera co-ordinate system is translated from the Earth co-ordinate system by $t_C^E$ then

$$\begin{aligned} x_i &= K R^C_{C_L} R_{EC} \left( X_i^E - t^E_{c_L} \right) \\ x_i' &= K R^C_{C_R} R_{EC} \left( X_i^E - t^E_{c_R} \right) \end{aligned} \tag{3-1}$$

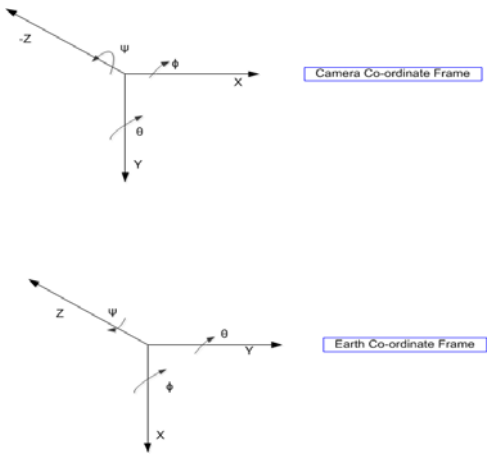


**Fig. 3-1** Co-ordinate Systems

The camera matrix K comes from camera calibration. In this research it is set to

$$k = \begin{pmatrix} 6178.88357 & 0 & 1372.57354 \\ 0 & 6133.64911 & 1071.34752 \\ 0 & 0 & 1 \end{pmatrix} \tag{3-2}$$

In this research if either camera is rotated about the x-axis, by $a_x$ radians, about the y-axis, by $a_y$ radians, and about the z-axis, by $a_z$ radians, in the camera co-ordinate system, then

$$R^C = R_z^C . R_y^C . R_x^C$$

$$R_z^C = \begin{pmatrix} \cos(a_z) & \sin(a_z) & 0 \\ -\sin(a_z) & \cos(a_z) & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$R_y^C = \begin{pmatrix} \cos(a_y) & 0 & -\sin(a_y) \\ 0 & 1 & 0 \\ \sin(a_y) & 0 & \cos(a_y) \end{pmatrix} \tag{3-3}$$

$$R_x^C = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(a_x) & \sin(a_x) \\ 0 & -\sin(a_x) & \cos(a_x) \end{pmatrix}$$

For each test case, n=20 3D points in the Earth co-ordinate system, $X_i^E = (x_i\ y_i\ z_i)^T$ , $1 \le n \le 20$, are generated in a quadrangle in the xy-plane with height equal to a value slightly divergent from a given depth value - for algorithmic reasons - as

$$\begin{aligned} x_i &= h_1 + (h_2 - h_1) * rand \\ y_i &= w_1 + (w_2 - w_1) * rand \\ z_i &= -d_1 - (d_2 - d_1) * rand \end{aligned} \tag{3-4}$$

where the w and the h values define the width and height of the xy-plane, the d values define the depth, and rand is a random number generator producing uniformly random numbers in the range $0 \le rand \le 1$. The mean value of $z_i$ is $-(d_2 + d_1)/2$ in Earth co-ordinates.

The equation (3-1) represents generation of image pixels. However, feature detection must follow [19],[20] and adds ±1 pixel Gaussian feature location noise. In this research this error is simulated with a Gaussian process with mean equal to the pixel position without noise and $\sigma = 3.291$ so that 99.9% of the noise will fall within ± 1 pixel of the mean.

Left and right camera image pixels may be processed to yield an estimated depth for each. An average depth can be calculated for the pair of images, an average for the test case can be easily calculated and, using [29] a confidence interval can be stated for image depth.

According to [29], the objective of statistics is to make inferences about a population based upon sample data. In the case of this research, uniformly distributed pixels are created and then processed in a sometimes linear and sometimes non-linear complex manner to produce estimated depths. The Central Limit Theorem implies that depth will be normally distributed.

For a 95% Confidence Interval (CI) $\alpha = 0.05$. When the sample size, n, is small, one may use Student's t distribution to construct CIs for the mean of a normal distribution [29] Thus for image depth, d, the 95% CI for the mean is

$$\bar{d} \pm t_{0.025} \frac{s}{\sqrt{n}} \tag{3-5}$$

It seems reasonable to require $t_{0.025} \le \sqrt{n}$ and this occur for $n \ge 7$. At n = 7, $t_{0.025}$ = 2.447, so we need 7 sample depths per test case.

# 4. Results

In the reported research, the following four basic test cases are run, differentiated in terms of the degree of camera placement error. All data is relative to the camera co-ordinate system. Image depth should be 2.0000 or 20.0000 meters, as shown

below. Actual results are given below in Table 5-1. It reports results without Gaussian pixel error and results with Gaussian pixel error.

1. x, y, z translation both cameras: no error; x, y, z rotation both cameras: no error. No image rectification.
2. x, y, z translation both cameras: no error; x, y, z rotation both cameras: no error. Image rectification.
3. x, y, z translation both cameras: left - 0.002m right - -0.002m; x, y, z rotation both cameras: $0.05^{o}$. Image rectification.
4. x, y, z translation both cameras: left - -0.002m right -0.002m; x, y, z rotation both cameras: $-0.05^{0}$. Image Rectification.

Within each of these tests are 8 sub-tests. For the first four:
Earth points: $w_1 = 0.02$, $w_2 = 0.06$; $h_1 = -0.03$, $h_2 = 0.2$; $d_1 = 1.9975$, $d_2 = 2.0025$ for (3-5).
For the last four:
Earth points: $w_1 = 0.02$, $w_2 = 0.06$; $h_1 = -0.03$, $h_2 = 0.2$; $d_1 = 1.9975$, $d_2 = 2.0025$ for (3-5).

1. Pose of left camera: (x, y, z): (-0.1m, 0.0m, 2.0m), ( 0.0°, 0.0°, 0.0°)
   Pose of right camera: (x, y, z): ( 0.1m, 0.0m, 2.0m), ( 0.0°, 0.0°, 0.0°)
2. Pose of left camera: (x, y, z): (-0.1m, 0.0m, 2.0m), ( 0.0°, -1.0°, 0.0°)
   Pose of right camera: (x, y, z): ( 0.1m, 0.0m, 2.0m), ( 0.0°, 1.0°, 0.0°)
3. Pose of left camera: (x, y, z): (-0.1m, 0.0m, 2.0m), ( 0.0°, -5.0°, 0.0°)
   Pose of right camera: (x, y, z): ( 0.1m, 0.0m, 2.0m), ( 0.0°, 5.0°, 0.0°)
4. Pose of left camera: (x, y, z): (-0.1m, 0.0m, 2.0m), ( 0.0°, -10.0°, 0.0°)
   Pose of right camera: (x, y, z): ( 0.1m, 0.0m, 2.0m), ( 0.0°, 10.0°, 0.0°)
5. Pose of left camera: (x, y, z): (-0.5m, 0.0m, 20.0m), ( 0.0°, 0.0°, 0.0°)
   Pose of right camera: (x, y, z): ( 0.5m, 0.0m, 20.0m), ( 0.0°, 0.0°, 0.0°)
6. Pose of left camera: (x, y, z): (-0.5m, 0.0m, 20.0m), ( 0.0°, -1.0°, 0.0°)
   Pose of right camera: (x, y, z): ( 0.5m, 0.0m, 20.0m), ( 0.0°, 1.0°, 0.0°)
7. Pose of left camera: (x, y, z): (-0.5m, 0.0m, 20.0m), ( 0.0°, -5.0°, 0.0°)
   Pose of right camera: (x, y, z): ( 0.5m, 0.0m, 20.0m), ( 0.0°, 5.0°, 0.0°)
8. Pose of left camera: (x, y, z): (-0.5m, 0.0m, 20.0m), ( 0.0°, -10.0°, 0.0°)
   Pose of right camera: (x, y, z): ( 0.5m, 0.0m, 20.0m), ( 0.0°, 10.0°, 0.0°)

# 5. Conclusions

It is the goal of the reported research eventually to accurately predict robotic pose in order to allow accurate navigation in the absence of Global Positioning System (GPS) or similar signals, using stereo camera-equipped Unmanned Aerial Vehicles (UAVs) or robotic devices in general. Accurate navigation is applied to

UAVs in an urban environment, to aerial refueling, and to space-based robotic vehicles engaged in space debris clearance operations.

Pose prediction using stereo cameras requires knowledge of the relative pose of the left and right cameras. This knowledge, obviously, will not be available without error. Test 1 assumes perfect knowledge without the use of image rectification; test 2 still assumes perfect knowledge, but now with image rectification; while test 3 and 4 assume, respectively, positive, and then negative errors in position and angle, with image rectification. Within each test there are 8 sub-tests with various y-axis camera rotations and x,y,z-axes camera translations.

The results are shown in table 5-1. The table shows the results with no pixel position error and clearly shows that without image rectification an accurate depth can be calculated only with both cameras pointed without any rotational angle through the y-axis. With image rectification and for the stated camera pose errors and up to 10° camera y-axis rotation, image depth can be known with less than 7.0% relative error.

Table 5-1 also shows the results for the same test cases, but with Gaussian pixel error that adds image depth confidence interval width variability. Now maximum depth error is less than 4.3% relative error. On the other hand, the width of the 95% confidence interval around the mean image depth is as much as 0.43m at 20m depth, or 2% relative variation in depth. Thus maximum relative error can be as much as about 6%.

The image depth chosen was restrictive, only 2m and 20m performance was studied. Future research should look at greater image depths, say 10, 50, and 100 meters. Also, image depth performance without image rectification should be examined for camera pose angles of less than 5°, say 0.5°, 1.0°, 2.0°, and 4.0°. It is possible that small pose errors can still give reasonable performance without using image rectification.

**Table 5-1** Actual Image Depths

| Test Number | Sub-test Number | No Pixel Error | | Gaussian Pixel Error | |
|---|---|---|---|---|---|
| | | Image Depth (m.) 95% CI | Error Image Depth (m.) 95% CI | Image Depth (m.) 95% CI | Error Image Depth (m.) 95% CI |
| 1 | 1 | 2.0001±0.0003 | -0.0001±0.0003 | 2.0001±0.0004 | 0.0001±0.0004 |
| 1 | 2 | 1.4810±0.0001 | 0.5190±0.0001 | 1.4810±0.0003 | 0.5190±0.0003 |
| 1 | 3 | 0.7239±0.0000 | 1.2761±0.0000 | 0.7239±0.0000 | 1.2761±0.0000 |
| 1 | 4 | 0.4378±0.0000 | 1.5622±0.0000 | 0.4378±0.0000 | 1.5622±0.0000 |
| 1 | 5 | 20.0001±0.0004 | -0.0001±0.0004 | 20.0033±0.0082 | -0.0033±0.0003 |
| 1 | 6 | 11.7677±0.0007 | 8.2323±0.0007 | 11.7681±0.0013 | 8.2319±0.0013 |
| 1 | 7 | 4.4323±0.0007 | 15.5677±0.0007 | 4.4322±0.0008 | 15.5678±0.0008 |
| 1 | 8 | 2.4705±0.0003 | 17.5295±0.0003 | 2.4706±0.0004 | 17.5294±0.0002 |
| 2 | 1 | 1.9998±0.0003 | 0.0002±0.0003 | 1.9995±0.0006 | 0.0005±0.0006 |
| 2 | 2 | 1.9999±0.0004 | 0.0001±0.0004 | 2.0000±0.0003 | 0.0000±0.0003 |
| 2 | 3 | 1.9999±0.0002 | 0.0001±0.0002 | 1.9998±0.0005 | 0.0002±0.0005 |
| 2 | 4 | 2.0001±0.0004 | -0.0001±0.0004 | 2.0003±0.0007 | -0.0000±0.0007 |
| 2 | 5 | 19.9999±0.0002 | 0.0001±0.0002 | 19.9979±0.0103 | 0.0021±0.0103 |
| 2 | 6 | 20.0001±0.0002 | -0.0001±0.0002 | 19.9954±0.0093 | 0.0046±0.0093 |
| 2 | 7 | 20.0000±0.0005 | -0.0000±0.0005 | 20.0022±0.0058 | -0.0022±0.0058 |
| 2 | 8 | 20.0001±0.0004 | -0.0001±0.0004 | 19.9946±0.0110 | 0.0054±0.0110 |
| 3 | 1 | 2.0706±0.0005 | -0.0706±0.0005 | 2.0745±0.0090 | -0.0745±0.0090 |
| 3 | 2 | 2.0707±0.0004 | -0.0707±0.0004 | 2.0764±0.0040 | -0.0764±0.0040 |
| 3 | 3 | 2.0710±0.0003 | -0.0710±0.0003 | 2.0821±0.0041 | -0.0821±0.0041 |
| 3 | 4 | 2.0710±0.0002 | -0.0710±0.0002 | 2.0756±0.0114 | -0.0756±0.0114 |
| 3 | 5 | 21.4985±0.0058 | -1.4985±0.0058 | 21.6108±0.1846 | -0.6108±0.1846 |
| 3 | 6 | 21.4914±0.0185 | -1.4914±0.0185 | 20.1190±0.4369 | -0.1190±0.4369 |
| 3 | 7 | 21.4715±0.0103 | -1.4715±0.0103 | 20.8610±0.4382 | -0.8610±0.4382 |
| 3 | 8 | 21.44583±0.0108 | -1.4583±0.0108 | 20.4718±0.3235 | -0.4718±0.3235 |
| 4 | 1 | 1.9334±0.0001 | 0.0666±0.0001 | 1.9271±0.0024 | 0.0728±0.0024 |
| 4 | 2 | 1.9334±0.0003 | 0.0666±0.0003 | 1.9271±0.0031 | 0.0729±0.0031 |
| 4 | 3 | 1.9330±0.0003 | 0.0670±0.0003 | 1.9274±0.0045 | 0.0726±0.0045 |
| 4 | 4 | 1.9331±0.0002 | 0.0669±0.0002 | 1.9239±0.0058 | 0.0760±0.0058 |
| 4 | 5 | 18.7049±0.0046 | 1.2951±0.0046 | 19.9721±1.0226 | 0.0279±1.0226 |
| 4 | 6 | 18.7086±0.0110 | 1.2914±0.0110 | 19.4199±0.1835 | 0.5801±0.1835 |
| 4 | 7 | 18.7322±0.0112 | 1.2678±0.0112 | 20.5253±1.1037 | -0.5253±1.1037 |
| 4 | 8 | 18.7365±0.0088 | 1.2635±0.0088 | 19.8085±0.7882 | 0.1915±07882 |

# References


1. M. Fravolini, G. Campa, A. Ficola, M. Napolitano, and B. Seanor, "Modeling and Control Issues for Machine Vision-Based Autonomous Aerial Refueling for UAVs," Journal of Guidance, Control, and Dynamics (unpublished).
2. J. Valasek, K. Gunnam, J. Kimmett, M. Tandale, J. Junkins, and D. Hughes, "Vision Based Sensor and Navigation System for Autonomous Air Refueling," Journal of Guidance, Control, and Dynamics, Vol. 28, No. 5, 2005, pp. 979-989.
3. T. Webb, R. Prazenica, A. Kurdial, and R. Lind, "Vision-Based State Estimation for Autonomous Micro Air Vehicles," Journal of Guidance, Control, and Dynamics, Vol. 30, No. 3, 2007, pp. 816-826.
4. K. Kaiser, N. Gans, and W. Dixon, "Localization and Control of an Aerial Vehicle through Chained, Vision-Based Pose Reconstruction," Proceedings of the American Control Conference, IEEE, 2007, pp. 5934-5939.
5. K. Kaiser, N. Gans, S. Mehta, and W. Dixon, "Position and Orientation of an Aerial Vehicle through Chained, Vision-Based Pose Reconstruction," Proceedings of the AIAA Guidance, Navigation, and Control Conference, AIAA-02006-6719, , AIAA, 2006.
6. John A. Volpe N.T.S. Center, "Vulnerability Assessment of the Transport Infrastructure Relying on the Global Positioning System," Report, Office of the Assistant Secretary for Transportation Policy, U.S. Department of Transportation, Aug. 2001.
7. Bachrach, S. Prentice, R. He, P. Henry, A. Huang, and M. Krainin, "Estimation, planning and mapping for autonomous flight using an RGB-D camera in GPS-denied environments", Int'l J. Robotics Research, Vol. 31(11), 2012, pp. 1320-1343.
8. F. Fraundorfer and D. Scaramuzza, 'Visual Odometry Part II: Matching, Robustness, Optimization, and Applications", IEEE Robotics and Automation Magazine, Vol. 19(2), 2012, pp. 78-90.
9. T. Kanade, O. Amidi, and Q. Ke, "Real-time and 3D Vision for Autonomous Small and Micro Air Vehicles," Proceedings of the IEEE Conference on Decision and Control, IEEE, 2004, pp. 1655-1662.
10. D. Nister, O. Naroditsky, and J. Bergen, "Visual Odometry for Ground Vehicle Applications," Journal of Field Robotics, Vol. 23, No. 1, 2006, pp. 3-20.
11. P. Torr, and A. Zisserman, "Feature Based Methods for Structure and Motion Estimation," Vision Algorithms: Theory and Practice: International Workshop on Vision Algorithms, edited by B. Triggs, A. Zisserman, and R. Szeliski, Springer-Verlag, Berlin / Heidelberg, 2000, pp. 278-294.
12. R. Hartley, and A. Zisserman, Multiple View Geometry in computer vision, 2'nd ed, Cambridge University Press, 2003, pp. 87-131.
13. G. Hu, W. MacKunis, N. Gans, W. Dixon, J. Chen, A. Behal, and D. Dawson, "Homography-Based Visual Servo Control With Imperfect Camera Calibration," IEEE Trans. Automatic Control, Vol. 54, No. 6, 2009, pp. 1318-1324.
14. E. Michaelson, M. Kirchhof, and U. Stilla, "Sensor Pose Inference from Airborne Videos By Decomposing Homography Estimates," Proceedings of the XXth ISPRS Congress, Technical Comission III, Istanbul, Turkey, 2004.
15. B. Jaehne, Digital image processing, 5'th revised and extended ed, Springer, 2001, pp. 208-211.
16. L. Iocchi, "Stereo Vision: Triangulation," URL: www.dis.uniroma1.it/~iocchi/stereo/triang.html.
17. M. Slonka, V. Hlavac, and R. Boyle, Image Processing, Analysis, and Machine Vision, 3'rd edition, Thomson, 2008, pp. 566.
18. C. Loop and Z. Zhang, "Computing Rectifying Homographies for Stereo Vision", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Vol. 1, IEEE Computer Society, 1999, pp. 125-131.

19. J. Sasiadek and M. Walker. “Vision-Based UAV Navigation,” Proceedings of the AIAA Guidance, Navigation and Control Conference, Honolulu, Hawaii, 2008.
20. J. Sasiadek and M. Walker, “Feature Detector Performance for UAV Navigation,” Proceedings of the IASTED Modelling, Identification and Control Conference, Innsbruck, Austria, 2010.
21. J. Sasiadek, M. Walker, and A. Krzyzak, “Feature Matching for UAV Navigation in Urban Environments,” Proceedings of the Conference on Methods and Models in Automation and Robotics, Miedzyzdroje, Poland, 2010.
22. J. Sasiadek, M. Walker, and A. Krzyzak, “Accurate Feature Matching for Autonomous Vehicle Navigation in Urban Environments,” Proceedings of the Conference on Methods and Models in Automation and Robotics, Miedzyzdroje, Poland, 2011.
23. M. Walker and J. Sasiadek, “Accurate Pose Determinatin for Autonomous Vehicle Navigation,” Proceedings of the Conference on Methods and Models in Automation and Robotics, Miedzyzdroje, Poland, 2013.
24. J. Bouguet, “Camera Calibration Toolbox for Matlab,” URL: www.vision.caltech.edu/bouguetj/calib_doc.
25. T. Strat, “Recovering the Camera Parameters from a Transformation Matrix,” Proceedings of the DARPA Image Understanding Workshop, New Orleans, LA, 1984, pp. 264-271.
26. R. Tsai, “A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses,” IEEE J. Robotics and Automation, Vol. RA-3, No. 4, 1987, pp. 323-344.
27. J. Heikkila, “Geometric Camera Calibration Using Circular Control Points,” IEEE Trans. Pattern Analysis and Machine Intelligenece, Vol. 22, No. 10, 2000, pp. 1066-1077.
28. Z. Zhang, “A Flexible New Technique for Camera Calibration,” IEEE Trans. Pattern Analysis and Machine Intelligenece, Vol. 22, No. 11, 2000, pp. 1330-1334.
29. W. Mendenhall, Introduction to Linear Models and the Design and Analysis of Experiments, Wadsworth Publishing Company, 1968, pp. 239-256.