A Comparison Between Homography and Direct Linear Transforms Algorithms for Unmanned Aerial Vehicle Automatic Landing

Muangmol Senpheng, and Siwakorn Kaewrat*

Department of Electrical and Electronic Engineering, Loei Rajabhat University, Loei, 42000, Thailand

Muangmol.sen@lru.ac.th, Siwakorn.kae@lru.ac.th*

Abstract. *The purposes of this research were to compare between homography and direct linear transformation (DLT) algorithms for estimating the distance between unmanned aerial vehicle and the markers on runway while automatic landing and to compare the accuracy of the actual distance .The experriment have two trials, First, the square was designed with a yellow square as 20 mm. take a picture with the distance between the center of the camera to the center of the a yellow square mark as 400 mm .and testing with homography and DLT algorithms, The homography algorithm detected is 20.38 mm. 1.9% error, DLT algorithm detected is 16.47 mm. 17.65% error. The result of finding the distance between the center of the camera to the center of the yellow square, homography detected algorithm 401.99 mm. 0.49% error and the DLT algorithm detected 687.2 mm. 71.82% error. Second, the square was designed with two green squares as 20 mm. and take a picture with the distance between the center of the camera to the center of the two green squares mark as 300 mm. Testing homography and DLT algorithms, The homography algorithm at marker detected is 21.27 mm. 6.35% error, at second marker detected is 21.93 mm. 9.65% error, DLT algorithm at first marker detected is 9.52 mm. 52.44% error, at second marker detected is 12.47 mm. 37.65% error. The result of finding the distance between the center of the camera to the center of the green square, homography algorithm first marker detected 311.02 mm. 3.6733% error, at second marker detected 296.57 mm. 1.1433% error and DLT algorithm at first marker detected 112.89 mm. 62.37% error, at second marker detected 125.74 mm. 58.066% error. In conclusion, a square yellow designed with homograph algorithm is more accurate than DLT algorithms.*

Received by 01 June 2023
Revised by 29 September 2023
Accepted by 03 October 2023

Keywords:

UAV, homography, DLT

1. Introduction

At present, Unmanned Aerial Vehicle: UAV has developed a lot and it was used to a variety of uses, such asspraying pesticides in agriculture, aerial photography for mapping, Live broadcast of the television station, tracking the suspects or criminals of the police department, surveillance on the border seams of the territorial defense agencies and etc. When we bring the equipment according to the purpose of each task install on Unmanned Aerial Vehicle (UAV), it can achieve the purpose of each job.

The researcher's purpose is to develop of Unmanned Aerial for Vehicle (UAV) mark detection, which can determine the position of the runway in three-dimensional coordinates by detecting the position of the mark on the airport runway. It can give the coordinates between the plane and the earth that can be positioned and also useful for automatic landing in the future.

2. Literature Review

This chapter is showing papers in automatic landing system using computer vision and the field of work in detecting and finding distance between the center of the camera to the center of markers on runway by homography and direct linear transform algorithms.

Muangmol et al., in 2017. proposed Detection of Markers on Runway for distance of Automatic Landing Using Computer Vision. The method is detection markers on runway by homography algorithm for distance of Automatic Landing [1].

Muangmol et al., in 2016 proposed the method of detection stripe lines and marker on runway by homography and direct linear transform for Automatic Landing [2].

Li Sha Sha, in 2013. proposed The Research of the Lane Detection Algorithm Base on Vision Sensor. This paper focus of the study, based on the visual structure of the road stressing on environment recognition [3].

King Hann LIM et al., in 2009 proposed Lane Vehlcle Detection and Tracking. This paper presents a monocular lane-vehicle detection and tracking system with comprising of lane boundary detection, lane region tracking, and vehicle detection. [4]

3. Proposed System

In this research, It have two methods are the one yellow square and the two green squares on general runway for the detection and find distance between from center of camera to center of square markers by Logitech camera with mimetic runway.The two methods use threshold images for squares detection and take homography and direct linear transform algorithms for find distance. Finally, comparision between homography and direct linear transform algorithms for center of square markers geometry is calculate compared with center of camera.

3.1.1 Projection and Camera Model

The perspective projection of observed world point onto an image point establish on a retinal plane. When make use of homogenous coordinates, relation of the projective between observed world point and an image point can be expressed.

When we consider the relationship of objects in 2D to the model of the camera, considering the X axis as shown in Figure 2, we have $x = f\frac{X}{Z}$ and $\frac{x}{f} = \frac{X}{Z}$

When we consider the relationship of objects in 2D to the model of the camera, considering the Y axis as shown in Figure 3, we have $y = f\frac{Y}{Z}$ and $\frac{y}{f} = \frac{Y}{Z}$

We get the relation of object in 2D to the camera model, considering the x-axis and y-axis is $x = f\frac{X}{Z}$ and $y = f\frac{Y}{Z}$. It will be noted that the Z-value of the equation is a divisor from (*X*, *Y)* coordinate to (*x, y)* coordinates and It is a nonlinear relative equation.

When $Z = Z_0$, we will get $Y = f\frac{Y}{z_0}$ and $X = f\frac{X}{z_0}$

3.1.2 Intrinsic Parameter

From equation $X = f\frac{X}{z_0}$ and, When the $Y = f\frac{Y}{z_0}$ pixels of the camera sensor do not a symmetrical square, it will be

$$X = kf\frac{X}{z} \quad \text{and} \quad Y = lf\frac{Y}{z} \qquad (1)$$

When

x, y = coordinates (pixel)

k, l = scale parameter (pixel/m)

f = focal length (m or mm)

$$f_x = kf \quad , \qquad f_y = lf \qquad (2)$$

$$x = f_x\frac{X}{Z} \ , \quad y = f_y\frac{Y}{Z} \qquad (3)$$

When image centre or principal point (c) does not at the origin point, The position of image center in images plane is c_x, c_y as equation *(*4*)*.

$$x = f_x\frac{X}{Z} + c_x \quad , \qquad y = f_y\frac{Y}{Z} + c_y \quad (4)$$

3.1.3 Extrinsic Parameter

On the axis of camera frame does not overlap with world frame that we can write camera frame and world frame as equation (5).

$$X^C = R_W^C X^W + T_W^C \qquad (5)$$

When

X^C = 3D point coordinates in camera frame

X^W = 3D point coordinates in world frame

R_W^C = rotation matrix of world frame in camera frame

T_W^C = translation matrix of world frame in camera frame

3.1.4 Threshold

In this research, we divide the image to find a rectangular point in the image by threshold .It's the easiest way to split an image .Because it does not complicate and the results are quite good .

$$g(x,y)=\begin{cases}1 & if \quad (x,y)>Th \\ 0 & otherwise\end{cases} \qquad (6)$$

When

$g(x,y)$ = Image data atposition x, y

Th = threshold value

3.1.5 Pose Estimation

It is calculation 3D position value of markers which compared to the camera and show in the form of 4x4 matrix, correlated between camera coordinated frame and world coordinated frame .

$$\begin{bmatrix} X_C \\ Y_C \\ Z_C \\ 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & T_1 \\ r_{21} & r_{22} & r_{23} & T_2 \\ r_{31} & r_{32} & r_{33} & T_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_M \\ Y_M \\ Z_M \\ 1 \end{bmatrix} \qquad (7)$$

When

X_M = real world coordinates in the X axis

Y_M = real world coordinates in the Y axis

Z_M = real world coordinates in the Z axis

X_C = camera coordinates in the X axis

Y_C = camera coordinates in the Y axis

Z_C = camera coordinates in the Z axis

x_d , y_d = observed in screen coordinate

x_c , y_c = observed in camera coordinate

The relationship between (X_C , Y_C , Z_C) points on camera coordinated frame with matching points in ideal screen coordinated frame according to Perspective projection as equation (8).

$$\begin{bmatrix} x_c \\ y_c \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & x_d \\ 0 & f_y & y_d \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_C \\ Y_C \\ 1 \end{bmatrix} \qquad (8)$$

3.1.6 Homography

Homography is relationship between the real world and image plane by the image of the object on the real world that project through the lens and crashed on the image sensor make an image of a real-world object on the image plane. It can be represented by the equation (9).

$$q \;=sWMQ \qquad (9)$$

$$Q=\begin{bmatrix} X \\ Y \\ 1 \end{bmatrix},\; M=\begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix},\; W=\begin{bmatrix} r_{11} & r_{12} & T_1 \\ r_{21} & r_{22} & T_2 \\ r_{31} & r_{32} & T_3 \end{bmatrix},\; q=\begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

When

q = point on the image plane

s = constant scale

H = homography

Q = point on the real world plane

M = intrinsic matrix

W = extrinsic matrix

This can be represented in the equation (10) and (11).

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = s\begin{bmatrix} r_{11} & r_{12} & T_1 \\ r_{21} & r_{22} & T_2 \\ r_{31} & r_{32} & T_3 \end{bmatrix} \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} \qquad (10)$$

$$\begin{bmatrix} X_1 & Y_1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ X_4 & Y_4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & X_1 & Y_1 & 1 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & X_4 & Y_4 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & X_1 & Y_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & X_4 & Y4 & 1 \end{bmatrix} \begin{bmatrix} h_{11} \\ h_{12} \\ h_{13} \\ h_{21} \\ h_{22} \\ h_{23} \\ h_{31} \\ h_{32} \\ h_{33} \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \qquad (11)$$

3.1.7 Direct Liner Transform

It is an algorithm for estimating the position and direction for the position of the camera directly on the floor by ignoring the terms of possible outcomes, all possible .The camera recording can be explained as shown in the Fig. 1 (a), it is object) point (O) mapping in object space to image point (I') on camera plane and make the digitization must do projected again into projected plane (I) as Fig. 1 (b)

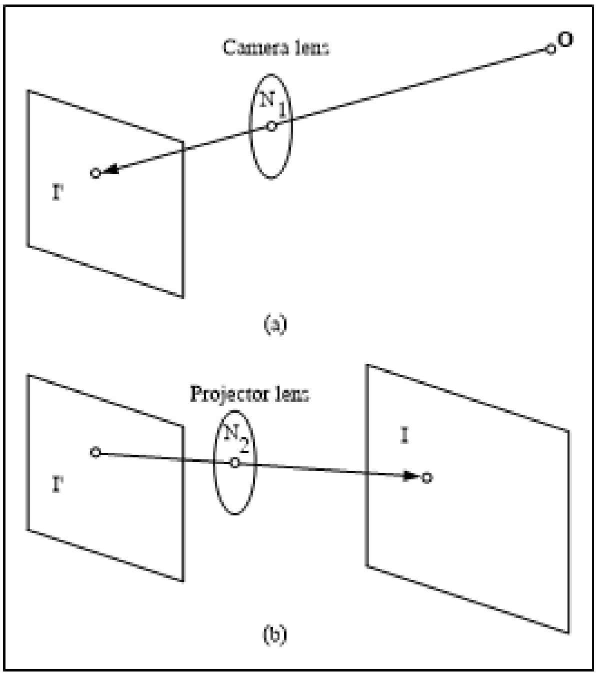

Fig. 1 (a) object (point O) mapping in object space to image point (I') on camera plane (b) digitization projected again into projected plane (I)

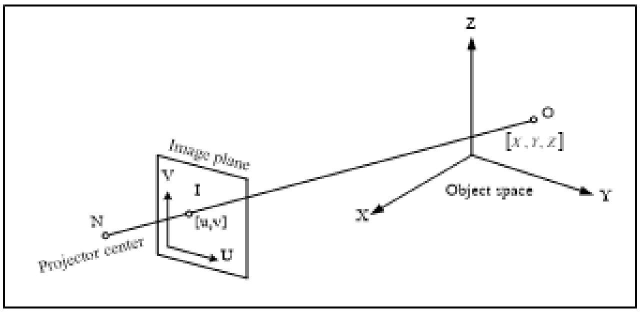

Fig. 2 The relationship between projected image and object

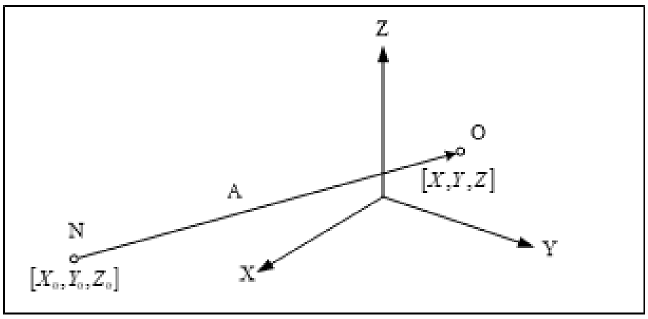

Fig. 3 The coordinates of projection center point

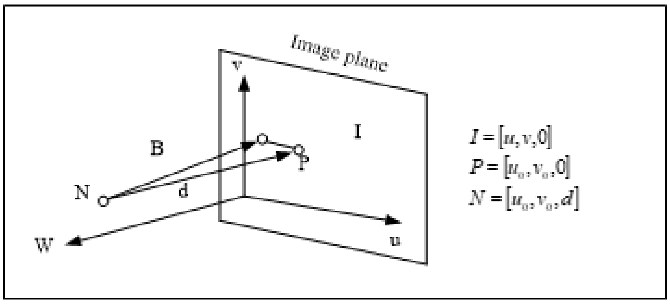

Fig. 4 Image plane 2D

We can write the equation of relationship between object point coordinates in the frame of object space and image point on image plane as equation (12), (13) and (14).

$$\bar{p}_n = sM_{ext}P^w \quad (12)$$

$$x = \frac{r_{11}X + r_{12}Y + r_{13}Z + t_x}{r_{31}X + r_{32}Y + r_{33}Z + 1}$$

$$y = \frac{r_{21}X + r_{22}Y + r_{23}Z + t_Y}{r_{31}X + r_{32}Y + r_{33}Z + 1} \quad (13)$$

$$r_{11}X + r_{12}Y + r_{13}Z + t_x - x(r_{31}X + r_{32}Y + r_{33}Z + t_Z) = 0$$

$$r_{21}X + r_{22}Y + r_{23}Z + t_y - y(r_{31}X + r_{32}Y + r_{33}Z + t_Z) = 0 \quad (14)$$

Apply equation) 14 (to write in the form of equation $AX = 0$ for find rotation and translation value and give $Z = 0$

$$AX = \begin{pmatrix} X & Y & 0 & 0 & 0 & 0 & -xX & -xY & 0 & 1 & 0 & -x \\ 0 & 0 & 0 & X & Y & 0 & -yX & -yY & 0 & 0 & 1 & -y \end{pmatrix} \begin{pmatrix} r_{11} \\ r_{12} \\ r_{13} \\ r_{21} \\ r_{22} \\ r_{23} \\ r_{31} \\ r_{32} \\ r_{33} \\ t_x \\ t_y \\ t_z \end{pmatrix} = 0$$

(15)

3.1.8 Implemention Example

We tested some part of the proposed algorithm using the following steps :we started from camera calibration, image undistorted, image threshold and detect markers by hough transform and direct linear transform algorithm.

4. Experimental Results

Result of runway testing using the yellow square with homography algorithm is illustrated in Fig. 5,

Fig. 5 Result of homography algorithm using 1 yellow square

Result of runway testing using the yellow square with direct linear transform algorithm. is illustrated in Fig. 6,

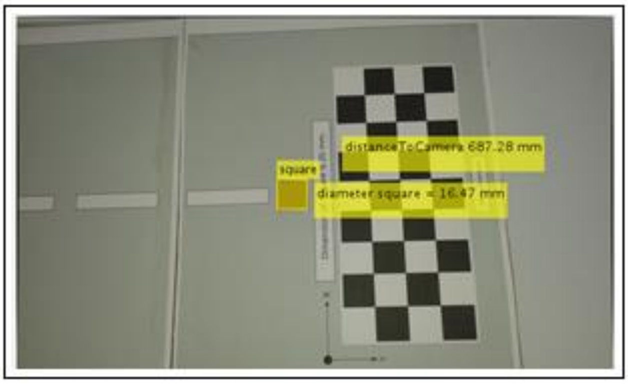

Fig. 6 Result of direct linear transform algorithm using 1 yellow square

World-Points	H		DLT	
X	57.300558	99.751606	-53.248317	126.65304
Y	57.981948	79.38271	-59.844677	141.74446

Table 1 Apply the inverse transformation from image to world

Compute the diameter of the square in millimeters	
H	DLT
20.38	16.47

Table 2 Compute the diameter of the square in millimeters

Compute the distance from center of the square to the camera in millimeters	
H	DLT
401.994336430604	687.28

Table 3 Compute the distance from center of the square to the camera in millimeters

Result of runway testing using 2 green squares with homography algorithm is illustrated in Fig. 7,

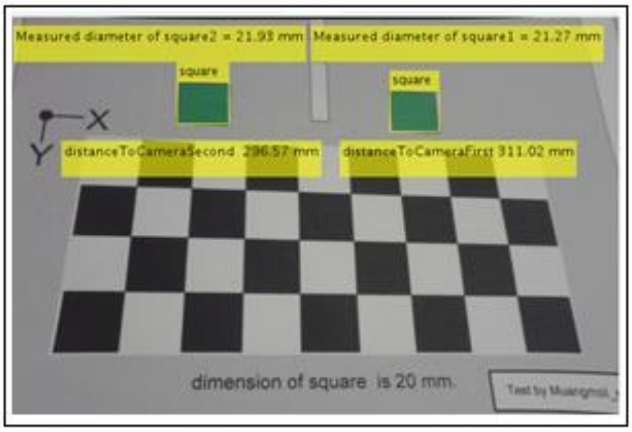

Fig.7 Result of homography algorithm using 2 green squares

Result of runway testing using 2 green squares with direct linear transform algorithm is illustrated in Fig. 8

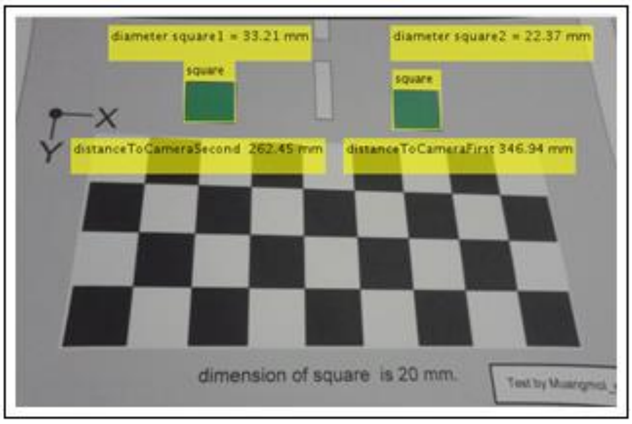

Fig. 8 Result of direct linear transform algorithm using 2 green squares

World-Points 1	H		DLT	
X	129.09920	88.26093	-29.172454	17.826584
Y	107.87330	89.66071	-31.859581	26.961099
World-Points 2	H		DLT	
X	40.330253	89.59564	-39.785487	60.020787
Y	18.440545	90.98238	-43.233035	72.008122

Table 4 Apply the inverse transformation from image to world

Measured diameter of square1	
H	DLT
21.27 mm	9.52 mm
Measured diameter of square2	
H	DLT
21.93 mm	12.47 mm

Table 5 Compute the diameter of the square in millimeters

	Compute the center of the square in the image 1			
	H		DLT	
Center1image	828	362.50	828	807.50
Center1world	117.9739	78.38314	-8.717144	26.371486
	Compute the center of the square in the image 2			
	H		DLT	
Center2image	828	807.50	1709	1693.50
Center2world	115.38656	31.21996	24.8001	77.66766

Table 6 Compute the center of the square in the image

Compute the distance from center of the first square to the camera	
H	DLT
311.02 mm	112.89 mm
Compute the distance from center of the second square to the camera	
H	DLT
296.57	125.74 mm

Table 7 Compute the distance from center of the square to the camera

5. Conclusion

In trial, the square was designed with a square yellow square as 20 mm. and take a picture with the distance between the center of the camera to the center of the one yellow square mark as 400 mm. Test of homography and DLT algorithms, The homography algorithm detected is 20.38 mm. 1.9% error, DLT algorithm detected is 16.47 mm. 17.65% error.

The result of finding the distance between the center of the camera to the center of the yellow square, homography detected algorithm 401.99 mm. 0.49% error and the DLT algorithm detected 687.28 mm. 71.8252% error.

In trial, the square was designed with two green squares 20 mm. and take a picture with the distance between the center of the camera to the center of the two green squares mark as 300 mm. and tested with Homography and DLT algorithms, The Homography algorithm marker one detected is 21.27 mm. 6.35% error, marker two detected is 21.93 mm. 9.65% error, DLT algorithm marker one detected is 9.52 mm. 52.4% error, marker two detected is 12.47 mm. 37.65% error .

The result of finding the distance between the center of the camera to the center of the green square, homography algorithm marker one detected 311.02 mm. 3.6733% error, marker two detected 296.57 mm. 1.1433% error and the DLT algorithm marker one detected 112.89 mm. 62.37% error, marker two detected 125.74 mm. 58.066% error.

In conclusion, a square yellow designed with homograph algorithm is more accurate than DLT algorithms.

6. References

[1] M. Senpheng, **"**Detection of markers on runway for distance of automatic landing using computer vision*," 9th ECTI-CARD*, Chiang khan, Thailand, 2017.

[2] M. Senpheng, "Detection of markers on runway and conversion to 3D information for automatic landing using computer vision," *Science and Technology RMUTT Journal*, vol. 6, no.1, pp. 181-198, 2016.

[3] L. S. Sha, "The Research of the Lane Detection Algorithm Base on Vision Sensor," *Research Journal of Applied Sciences, Engineering and Technology*, vol. 6, no.4, pp. 642-646, 2013.

[4] K. H. Lim, L. M. Ang, K. P. Seng, and S. W. Chin, "Lane Vehicle Detection and Tracking," *International Muti Conference of Engineers and Computer Scientists*, Hong Kong, Mar. 18 – 20, 2009.

Biographies

Muangmol Senpheng currently Department of Electrical and Electronic Engineering, Loei Rajabhat University, Loei, 42000, Thailand, has research interests in robotic, image processing and automatic control.

Siwakorn Kaewrat currently Department of Electrical and Electronic Engineering, Loei Rajabhat University, Loei, 42000, Thailand, has research interests in Digital Electronics Devices, Sensor&Transducer and Image Processing.