Military Technical College
Kobry El-Kobbah
Cairo, Egypt

10th International Conference
On Aerospace Sciences&
Aviation Technology

A New Model for Recognizing A Three-Dimensional Object Using Hu and Zernike Moment invariants

GOUDA* I. SALAMA SAYED** F. BAHGAT Nazir*** Hassan Elmhrez

Abstract

In this paper, we present a new model to recognize a three-dimensional (3D) object, which uses six Hu moment invariants and six Zernike moments as a feature vector. It extends our former work on 3D-object recognition based on Hu moment invariants [1].
The classification of the desired 3D object using Hu-moment model [1] is based on selecting the object that corresponding the minimum distance of the six minima obtained from six reference libraries and the desired object. The classification may be correct if we selected the object that corresponding the second minimum distance value instead of the first value. To select the correct one, this paper describes another model to recognize a 3D object based on Hu and Zernike moment invariants as a feature vector. Zernike moment invariants are used to find the pose of the aircraft at first to know from which library we can make the decision.
The proposed model differs significantly from many recent 3D recognition models, which emphasize on stereo reconstruction and structured light analysis. Several trajectories for 3, 6, and 9 aircraft are generated, using 3D Studio Max software. To study the performance of the proposed model, the aircraft patterns were chosen to test the proposed model because the views of these patterns are relatively difficult due to the similarity of their images.
Finally, we show that the proposed model achieves high recognition rates compared with that of the View Information Encoded With Network model (VIEWNET) [2] and the Hu moment invariants [1] using the approximately the same set of objects and using the same decision rule.

Key Words

Feature Extraction, Hu Moment invariants, Zernike Moments, 3D-object recognition

* Egyptian Armed Forces
** Ain-Shams University
*** Syrian Armed Forces

1. Introduction

The problem of 3D object recognition is one of the most challenging sub problems in computer vision. The difficulty in 3D object recognition is the potential variability in the images of an object at different view angles, i.e. only one side of an object can be seen from any given viewpoint, which is some times insufficient to recognize similar objects from each other. Another difficulty in 3D object recognition is the handling of 3D objects allowing additional degree of freedom for the orientation of the object in space. Another difficulty in 3D object recognition appears when the desired objects may occlude each other.

For the mentioned difficulties, all researchers assume that the 3D object is allowed one degree of freedom for rotation (rotation about vertical axis) or two degree of freedom. This paper describes a novel model to recognize 3D objects taking into account the 3D objects are allowed six degrees of freedom for rotation and translation as in real world. In this model, the image containing the desired object is first captured using 3D Studio MAX software. Then, preprocessing operations: binarization is applied. A set of statistical features (Hu-moment invariants and Zernike moments) is extracted from the geometry of the object. In the training phase, two types of library are built: one stores the Hu moment features references, and other stores the Zernike moment invariants as feature vector references. In the recognition phase, the Zernike moment invariants are used to determine the pose of aircraft to know from which Hu-library we can make the decision. Finally, accumulating evidence from sequences of 2D views until 3D recognition is assured.

There are a variety of approaches for attacking the 3D object recognition problem. These include the approaches of Richard and Hemami [3], who used Fourier descriptors, and of Dudani et. al. who used moment invariants [4]. Tsai and Huang [5] solved the problem of determining the attitude of a 3D object given a minimum set of point correspondence. There is a different class of other approaches, which rely on 3D data (range data) for building 3D models of the objects to be identified [6]. We will not be concerned with this approach since we assume only image data from a single camera. There have been many studies on technology of recognizing a 3D object from its 2D image. As example, parametric eigenspace method [7]; neural networks [8, 9]; self-organizing neural architecture called VIEWNET [2]; and modular neural networks [10].

Section 2 describes a proposed 3D model to recognize 3D objects based on the Hu and Zernike moment invariants. The definition of the Hu moment invariants and Zernike moments are introduced in section 3. The training phase is illustrated in section 4. Section 5 illustrates the recognition phase. Section 6 presents experimental results, in different situations including noise effect, to show the efficacy of the proposed model. Finally, section 7 presents concluding remarks and recommendation for future work.

2. Proposed model: overview

The unknown object, represented by the extracted Hu invariant moments ϕ_j, is identified by matching ϕ_j with reference moments ϕ_j^I stored in six libraries using Euclidean distance and finding the minimum distance in each library. Then, sorting these six minima in ascending order. The index of the first minimum distance

indicates the type of the aircraft. So, the decision of the model described in [1] is based on selecting the first minimum of sorting six minima. But, from optimization point view, this rule may be not correct. In other words, since these six minima are found as a distance, so the difference between first and second minima of sorting six minima may be very small and the decision may be correct if the second sorted minima is selected instead of the first sorted minima. Therefore, the subject must be addressed: the selection of candidates from these six minima.
To select these candidates, the model described in [1] is modified by finding the pose of the aircraft at first to know from which library the decision could be taken. In order to achieve this objective, additional features must be added to the Hu invariant moments. Zernike moments with Hu invariant moments have been used as a feature vector. Zernike moments are used to find the pose of aircraft at first to know which library we can take the decision.

3. Feature extraction

The feature extraction aspect of image analysis seeks to identify inherent characteristics, or features, of objects found within an image. These characteristics are used to describe the object, or attributes of the object, prior to subsequent task of classification or recognition. Feature extraction operates on 2D image arrays but produces a list of descriptions, or a feature vector. These descriptions, or feature vector should be invariant to position, orientation, and scale aspects of the object. The definition of the Hu moment invariants and Zernike moments are introduced next

3.1 Hu moment Invariants

For a digital image, stored in a 2D array, the moment of order ($p + q$) is given by [11]:

$$m_{pq} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} x^p y^q f(x,y), \qquad p,q = 0,1,2,\ldots,\infty \qquad \textbf{(1)}$$

where M, N are the horizontal and vertical dimensions respectively of the image, $f(x,y)$ is the intensity (grey level) at a point (x,y) in the image.
From Eq. (1), it should be noted however that these basic moments are limited in their usefulness since they vary according to their position with respect to the origin and the scale and orientation of the object under investigation [11]. A set of invariant moments would be of more use. These can be derived by first calculating the central moments, μ_{pq}, with respect to the centroid, as given by [11]:

$$\mu_{pq} = \sum_{x} \sum_{y} (x-x')^p (y-y')^q f(x,y) \qquad (2)$$

Then developing the normalized central moments, η_{pq}, as

$$\eta_{pq} = \frac{\mu_{pq}}{(\mu_{00})^{\lambda}} \qquad (3)$$

where $\lambda = \frac{(p+q)}{2} + 1$, and $(p+q) \geq 2$.
These normalized central moments are invariant under translation and scaling aspects.
From these normalized parameters a set of invariant moments, $\{\phi\}$, may then be defined as follows [11]:

$$\phi_1 = \eta_{20} + \eta_{02} \qquad (4)$$

$$\phi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \quad (5)$$

$$\phi_3 = (\eta_{30} - 3\eta_{12})^2 + (\eta_{21} - \eta_{03})^2 \quad (6)$$

$$\phi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \quad (7)$$

$$\phi_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})\left[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2\right] + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})\left[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right] \quad (8)$$

$$\phi_6 = (\eta_{20} - \eta_{02})\left[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \quad (9)$$

This set of invariant moments makes a useful feature vector for the recognition of objects which must be detected regardless of position, size or orientation [11].

3.2 Zernike moments

The Zernike moments are related to the usual moments μ_{pq}. Thus once μ_{pq} have been calculated, we have the Zernike moments also. The equations of Zernike moment invariants expressed in terms of geometric moments are given as following [12]:

$$Z_1 = 3\left[2(\mu_{20} + \mu_{02}) - 1\right]/\pi \quad (10)$$

$$Z_2 = 9\left[(\mu_{20} - \mu_{02})^2 + 4(\mu_{11})^2\right]/\pi^2 \quad (11)$$

$$Z_3 = 16\left[(\mu_{03} - 3\mu_{21})^2 + (\mu_{30} - 3\mu_{12})^2\right]/\pi^2 \quad (12)$$

$$Z_4 = 144\left[(\mu_{03} + \mu_{21})^2 + (\mu_{30} + \mu_{12})^2\right]/\pi^2 \quad (13)$$

$$Z_5 = \frac{13824}{\pi^4}\{(\mu_{03} - 3\mu_{21})(\mu_{03} + \mu_{21})\} \times \left[(\mu_{03} + \mu_{21})^2 - 3(\mu_{30} + \mu_{12})^2\right] - (\mu_{30} - 3\mu_{12})(\mu_{30} + \mu_{12}) \times \left[(\mu_{30} + \mu_{12})^2 - 3(\mu_{03} + \mu_{21})^2\right] \quad (14)$$

$$Z_6 = \frac{864}{\pi^3}\{(\mu_{02} - \mu_{20})\left[(\mu_{03} + \mu_{21})^2 - (\mu_{30} + \mu_{12})^2\right] + 4\mu_{11}(\mu_{03} + \mu_{21})(\mu_{30} + \mu_{12})\} \quad (15)$$

These Zernike moments and Hu moments are extracted or calculated for each training image to form the feature vectors, used to construct the corresponding library.

4. Training phase

As flowcharted in Figure (1), the training phase consists of three steps: image preprocessing, feature extraction, and construction of libraries. Binarization of the input image is defined as a preprocessing step, followed by the extraction of six Hu invariant moments and six Zernike moments that are referred to as a two sub-vectors that form the extracted feature vector. Then these two sub-feature vectors are used in construction of the corresponding reference library.

The two types of libraries are as follows: the first one stores the feature vectors extracted from training images using Hu invariant moments, and the other one stores the feature vectors using Zernike moments. Each type of libraries consists of six libraries, which are created for each 3D aircraft for training, (X-library of dimension 90x6, Y-library of dimension 90x6, Z-library of dimension 90x6, XY-library of dimension 90x6, XZ-library of dimension 90x6, and YZ-library of dimension 90x6). Figure (2) represents how the feature vector consisting of six Zernike moments is stored in X-library. Similarly, there are another five libraries like X-library where six Zernike moments stored for training image samples rotated around Y-axis, Z-axis, XY-plane, XZ-plane, and YZ-plane. Another group of six libraries store six Hu moments as a feature vector for each training image.

The 3D objects that we used for the three-pattern experiment included F16, F18, and global hawk aircrafts. An example of the trained patterns shown in Fig. 4 We choose these three patterns because the views of these patterns are relatively difficult due to the similarity of their images.

5. Recognition phase

To demonstrate how the Hu-Zernike invariant moments- based object recognition model works, the flowchart of this model is illustrated in Figure (3). The first test image frame is transformed into binary image. From the resulting binary image, Hu-feature vector which contain the six Hu invariant moments, and Zernike-feature vector which contain the six Zernike invariant moments are extracted.

Hu-feature vector is matched with references feature vectors stored in six Hu moments libraries (X-library, Y-library, Z-library, XY-library, XZ-library, and YZ-library) using an Euclidean distance. In each library, the closest stored image sample to the input test image is found by choosing the minimum distance D_I, the index I of which will indicate the aircraft type. So, we obtain six minimum distances from six libraries as *minHdx*, *minHdy*, *minHdz, minHdxy, minHdxz,* and *minHdyz* from X, Y, Z, XY, XZ, and YZ libraries respectively. Also, we obtain six corresponding indexes Ix, Iy, Iz, Ixy, Ixz, and Iyz each of, which will indicate the aircraft type determined in its library. Next, the resulting six minima are sorted in ascending order. Then the corresponding decisions are sorted too. To explain this, Let, for example, we obtain the following decisions from six libraries as: from X library-F18, from Y library-F16, from Z library-mig21, from XY library-F18, from XZ library-kfir, and from YZ library-F16. Suppose that the sorted six Hu- minimum distances are as:
[minHdy, minHdx, minHdyz, minHdxz, minHdz, minHdxy],
then the corresponding decisions will be sorted as:
[F16, F18, F16, kfir, mig21, F18].

Similarly, Zernike-feature vector is compared with the references Zernike moment's libraries using an Euclidean distance D_{II}. But the output of each library is the minimum distance *minZd*, the index II which indicates the library name. In other words, this index indicates the pose of aircraft. Therefore, from X library we get the minimum distance *minZdx* and the corresponding index IIx, from Y library- *minZdy* and IIy, from Z library- *minZdz* and IIz, from XY library- *minZdxy* and IIxy, from XZ library- *minZdxz* and IIxz, and from YZ library- *minZdyz* and IIyz. These six minimum distances are sorted in ascending order. Basing on this sort, the corresponding indexes are sorted too. For example, suppose that the six zernike minimum distances are sorted as follows:
[minZdx, minZdy minZdxz, minZdz, minZdyz, minZdxy],
then the corresponding indexes will be sorted as:
[IIx, IIy, IIxz, IIz, IIyz, IIxy],
i.e. the sorted poses are as follows:
[X, Y, XZ, Z, YZ, XY].

Next, we take the first library name of sorting poses, which in our example is X. Then, finding the decision from Hu X library (in our example, it will be F18). The next step is to compare this decision with the first one of sorted six decisions, which is F16 in our example. If they are the same, the final decision for the first input test image was determined as target (1). In other case (as in our example), these two decisions

are different and we thus take the second library name of sorted poses (in our example-Y) and taking the decision from this library using Hu invariant moments (in our example-F16). This decision is compared with the second sorted decision (in our example-F18).

Again, if these two decisions are the same, the final decision for the first input test image was determined as target (1). If they are different (as it in our example), then we apply the following step. Normalizing the first minimum distance in both Hu and Zernike moment libraries by dividing them by the maximum value of six minimum distances, i.e.:

$$dH = \frac{\min\ (mindx,\ mindy,\ mindz,\ mindxy,\ mindxz,\ mindyz\ )}{\max\ (mindx,\ mindy,\ mindz,\ mindxy,\ mindxz,\ mindyz\ )} \quad (16)$$

$$dZ = \frac{\min\ (minzdx,\ minzdy,\ minzdz,\ minzdxy,\ minzdxz,\ minzdyz\ )}{\max\ (minzdx,\ minzdy,\ minzdz,\ minzdxy,\ minzdxz,\ minzdyz\ )} \quad (17)$$

The resulting normalized distances dH and dZ are compared to find the smallest.

Now, if dH is the smallest, then the final decision target (1) for the first input test image is determined as the first decision of sorted six Hu-decisions (in our example-F16). If dZ is the smallest, we take the decision from that Hu library, which is the first name in sorted six poses (in our example, this library is X and its decision is F18), as the final decision target (1).

Therefore, by recognizing the aircraft as target (1) (let it is F16), we arrive at the first evidence from first test frame. Applying the same processing on the second input frame of test image, we recognize the aircraft as target (2) (let it is F18) as the second evidence for second test frame.

Now, the two-evidence target (1) and target (2) are compared. If they are the same, then the identification of 3D aircraft is achieved as target (1). But if target (1) differs from target (2) as shown in our example, the third test frame is needed to make our decision for identification of the aircraft.

As we determine the aircraft name target (3) from third test frame (let it will be F18), we compare it with first and second evidence. First, target (1) and target (3) are compared. If they are the same, the identification of the 3D aircraft is represented as target (1). Otherwise, we compare target (2) with target (3). Again, if they are the same (as it in our example), the identification of the 3D aircraft is represented as target (2)(in our example-F18). Finally, if target (2) and target (3) are different, we can not identify the aircraft. By this way, we can identify the 3D aircraft if two from three evidence are identical.

6. Experiments and results

In this section, a number of experiments were conducted to compare the proposed 3D-recognition model with the Hu moment-based model and VIEW net model. The performance of the proposed model as the number of objects varies, and the rotation of the object around X-, Y-, and Z-axes varies, is illustrated.

Figure (6) shows the comparison among the proposed model, the Hu-moment model, and the VIEWNET model using approximately the same set of objects (3 aircrafts F16, F18, global hawk) for the same case of objects rotation around the vertical axis (Z-axis), i.e. one degree of freedom for rotation, and using the same decision rule, based on one 2D view (K=1) or successive of three 2D views (K=3).

Figure 7 shows example of the test patterns for the three types of aircraft that follows the trajectory shown in Fig. 3. To study the performance of the proposed model, a several trajectories for 3, 6, and 9 aircrafts are generated using 3D Studio Max software. Table 1 describe the first trajectory set that follow the ARC trajectory shown in Figure 3 at different angle of rotation about Y-axis from 0° to 5°, from 0° to 10°, from 0° to 15°, and from 0° to 20°, while rotation about Z-axis from -90° to -180° and X-axis is fixed 0

From Fig (8), it could be noticed that the results of the proposed model are better than the results of the Hu-moment model in case of using three objects. As shown in Fig. (8.a), the worst result yielded by Hu-moment model is 79% for arc1-5, while it 92.33% for the second model using only one 2D view. Similarly, using three successive 2D views (K=3)(see Fig. (8.b)), where the worst recognition rate is 80% in Hu-moment model, and 93.33% in the proposed model. Similarly, the proposed model improved the worst case in the Hu-moment model from 77.67% to 93% using frame step L=15 (see Fig. (8.c).

In case of six objects, the recognition rates in the proposed model are better than in the Hu-moment model using only one 2D view K=1 (see Fig. (9.a)), using three successive 2D views K=3 (see Fig. (9.b), and using three 2D views with frame step L =15 (see Fig. 9.c). Using K=1 and K=3, the proposed model improved the results obtained in the Hu-moment model from 69.67% (the worst result) to 86.83% at K=1, and from 70.83% to 88.5% at K=3. Similarly, using frame step L=15, the worst recognition rate in Hu-moment model is 67% for rotation about Y-axis 0°-20°, while it is 80% in the proposed model.

Table 2 describes another set of trajectories, which used to study the performance of the proposed model in case of rotation about three axes. The first where the object is rotated around Z-axis from -90° to -180°, X-axis from 0° to 5°, and Y-axis from 0° to 5°, i.e. the rotation about X- and Y- axes are performed with the same rate.

The proposed model was compared with the Hu-moment model using the test patterns for the three types of aircraft shown in Figure 7. Figure 10 plots the recognition rate versus the rotation angle about the X- and Y-axes when angle varies from 0° to 5°, from 0° to 10°, from 0° to 15°, and from 0° to 20° using one 2D view, three successive 2D views, and three 2D views with frame step L=15. Using K=1 and K=3, the proposed model improved the results obtained in the Hu-moment model from 86% (the worst result) to 95% at K=1, and from 88% to 95% at K=3. Similarly, using frame step L=15, the worst recognition rate in Hu-moment model is 91% for rotation about X-axis and Y-axis 0°-20°, while it is 100% in the proposed model.

From Figure 11, it could be noticed that the results obtained from the proposed model are better than the Hu-moment model in case of using six objects. Using K=1 and K=3, the proposed model improved the results obtained in the Hu-moment model from 84% (the worst result) to 92% at K=1, and from 85% to 93% at K=3. Similarly, using frame step L=15, the worst recognition rate in Hu-moment model is 90% for rotation about X-axis and Y-axis 0°-20°, while it is 97% in the proposed model.

Again, the proposed model was compared with the Hu-moment model for the case of using nine objects. Figure 12 plots the recognition rate versus the rotation angle about the X- and Y-axes when angle varies from 0° to 5°, from 0° to 10°, from 0° to 15°, and from 0° to 20° using one 2D view, three successive 2D views, and three 2D

views with frame step L=15. Using K=1 and K=3, the proposed model improved the results obtained in the Hu-moment model from 76% (the worst result) to 84% at K=1, and from 79% to 84% at K=3. Similarly, using frame step L=15, the worst recognition rate in Hu-moment model is 81% for rotation about X-axis and Y-axis 0°-20°, while it is 87% in the proposed model.

For the case of rotation about three axes, another sets of trajectories are generated. The first trajectory with 150 image frames for each object are generated where object is rotated around Z-axis -90° to -180°, X-axis from 0° to 5°, and Y-axis from 0° to 10°, i.e. the difference between the rotation about X-axis and rotation about Y-axis is 5°. Figure 13 plots the recognition rate versus the difference between the rotation about X-axis and rotation about Y-axis using one 2D view, three successive 2D, and using three 2D views with frame step L=15. The model was tested with another trajectories at different values of the difference between the rotation about X-axis and rotation about Y-axis (10°, 15°, 20°).

Using K=1 and K=3, the proposed model improved the results obtained in the Hu-moment model from 84% (the worst result) to 93% at K=1, and from 85% to 94% at K=3. Similarly, using frame step L=15, the worst recognition rate in Hu-moment model is 79% for rotation about X-axis and Y-axis 0°-20°, while it is 90% in the proposed model.

To study the effect of the noise on the proposed model, we choose several types of noise like ripple, diffuse, and wind. Figure 14 shows example images from the test patterns affected by ripple noise with 100, 300, and 400 in case of rotation about vertical axis (Z-axis). The recognition rate versus ripple values is plotted using one 2D view (see Fig. 14.a), using three 2D views with K=3 (see Fig. 14.b), and using three 2D views with L=15(see Fig.14.c).

For six objects tested for ripple noise, Using K=1 and K=3, the proposed model improved the results obtained in the Hu-moment model from 87% (the worst result) to 93% at K=1, and from 89% to 94% at K=3. Similarly, using frame step L=15, the worst recognition rate in Hu-moment model is 94% for rotation about X-axis and Y-axis 0°-20°, while it is 96% in the proposed model.

7. Time and storage measures

On objects with 541 image frame per object for training, the following time and storage measures has been observed for the first proposed model implemented using MATLAB (a sixth generation version) running under Microsoft Windows 98 as an operating system on a PC platform with the following configuration:

- Processor: Intel Pentium III processor 450 MHZ.
- Cache Memory: 512 kB cache.
- Memory: 64 MB RAM.

The time required to train a library with 90 image frames was 3527 seconds and 3723 seconds required to train a library with 91 image frames, i.e. 21358 seconds required to train an object with 541 image frames.

To conclude, for three object case, the training phase required a total of 64074 seconds, while the testing phase requires on average time of 35.81 seconds using only one 2D view and of 65.47 seconds or of 98.81 seconds using three 2D views.

8. Conclusion and Future work

The decision of the Hu moment model [1] is based on selecting the first value of sorting six minima obtained from six libraries, which may not be always correct. The decision may be correct if we select the second value instead of the first. To select the correct one, this paper describes a novel model to recognize a 3D object based on Hu and Zernike moment invariants as a feature vector. Zernike moment invariants are used to find the pose of the aircraft at first to know from which library we can make the decision.

Recognition is done by comparing an object against many representations stored in a library, and finding the closest match using Euclidean distance classifier. The decision of the proposed model is based on accumulating evidence from a set of three 2D views of the desired objects at different frame step (L), where L = 5, 7, 10, 15.

The proposed model was tested using several experiments using aircraft patterns, in different situations including noisy image, to show the efficacy. This model is robust against several types of noise like ripple, diffuse, and wind. Also, the performance of the proposed model was compared with that of the Hu moment model [1] and the VIEWNET model [2]. Using approximately the same set of objects (3 aircrafts F16, F18, global hawk) for the same case of objects rotation around the vertical axis (Z-axis) and using the same decision rule, which based on one 2D view or successive of three 2D views. The VIEWNET model yielded recognition rate of 90% with one view and up to 98.5% with three 2D views. In the Hu moment model, a recognition rate is up to 96.33% with one view, 98.67% with three 2D views, and up to 100% using frame step L=15. While the recognition rates obtained from the proposed model are of 99.33% with one view, 100% with three 2D views, and up to 100% using any frame step L (5, 7, 10, and 15).

References

[1] Sayed F. Bahgat, Gouda I. Salama, M. Elsaid Ghoneimy, and Nazir H. El mehrez, "Three-dimensional object recognition based on Hu Moment Invariants", Proceeding of the IASTED International Conference on Signal Processing, Pattern recognition, & Applications, June , 25-28, 2002, Crete, Greece.

[2] G. Bradski and S. Grossberg, "Fast learning VIEWNET architectures for recognition 3-D object from multiple 2-D views", Neural Networks, pp. 1053-1080, Vol. 8, No. 7, 1995.

[3] C. W. Richard and H. Hemami, "Identification of three-dimensional objects using Fourier descriptors of boundary curve", IEEE Trans. on Pattern Analysis and Machine Intelligence, pp.127-136, Vol. PAMI-2, 1980.

[4] S. A. Dudani; "Moment methods for the identification of three-dimensional objects from optical images", M. S. thesis, Ohio State Univ., Columbus, OH, 1971.

[5] R. Y. Tasi and T. S. Huang, "Uniqueness and estimation of three-dimensional motion parameters of rigid objects with curved surfaces", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. PAMI-6, 1984.

[6] T. L.-Perez and W. E. L. Grimson, "Recognition and location of overlapping parts from sparse data", Proc. of the 2nd Int. Symp. On Robotics Research, Kyoto, Japan, August 1984.

[7] H. Murase and S. K. Nayar, "Three-dimensional object recognition from appearance parametric eigenspace method" , Systems and Computers in Japan, pp. 45-53, Vol. 26, No. 8, 1995

[8] S. K. Nayar and H. Murase, "Visual learning and recognition of 3D objects from appearance" , International Journal of Computer Vision, pp. 5-24, Vol. 14, No. 1, 1995.

[9] T. Poggio and S. Edelman, "A network that learns to recognize three-dimensional objects", Nature, pp. 263-266, Vol. 343, 1990.

[10] H. Ando, S. Suzuki and T. Fujita, "Unsupervised visual learning of three-dimensional objects using a modular neural network architecture", Neural Network, pp. 1037-1051, Vol. 12, 1999.

[11] S. Dudani, K. Breeding and R. McGhee, "Aircraft identification by moment invariants", IEEE Trans. on Computers, pp. 39-45, Vol. 26, No. 1, 1977

[12] R. Mukundan and K. R. Ramakrishnan, "Moment functions in image analysis: theory and applications", World Scientific Publishing Co. Pte. Ltd. Singapore, 1998.

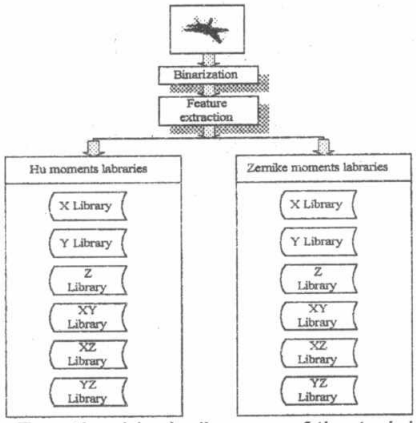


Fig. 1. Function block diagram of the training phase of the proposed model.

Object	Image index	Image pose, angle	Feature vector $Z_1\ Z_2 .\ Z_6$
1	1 2 • • 91	0° 1° • • 90°	
• •	• •	• •	
N	91x(N-1)+1 91x(N-1)+2 • • 91x(N-1)+91	0° 1° • • 90°	

Fig. 2. Feature vector representation in X library.

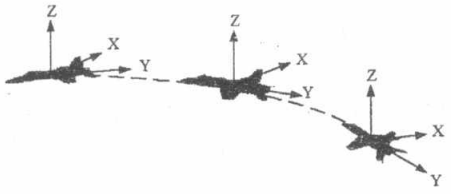


Fig. 3. Arc trajectory

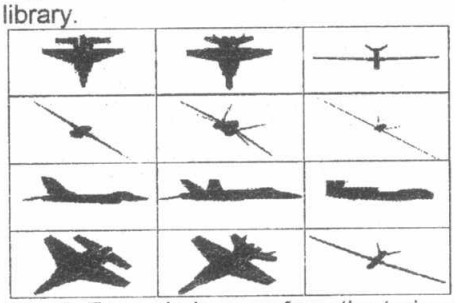


Fig. 4. Example images from the trained patterns:F16, (b) F18, (c) global hawk

	One 2D view, K=1	Three 2D views, K=3	Three 2D views with frame step L			
			5	7	10	15
VIEWNET model	90%	98.5%				
Hu moment based model	96.3%	98.7%	99 %	98.7 %	99.3 %	100 %
Hu-Zernike moment based model	99.3%	100%	10 0%	100%	100%	100 %

Fig. 6. Comparesion among the proposed model, Hu-moment model and VIEWNET model for rotation around vertical Z-axis.

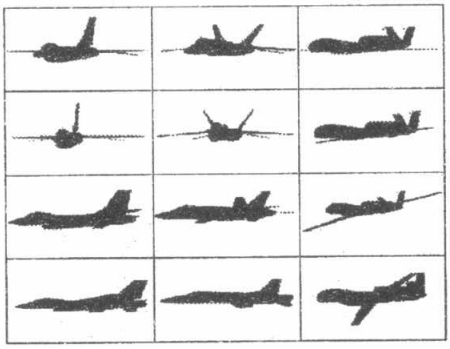


Fig. 7. Example images from the test patterns:(a) F16, (b) F18, (c) global hawk

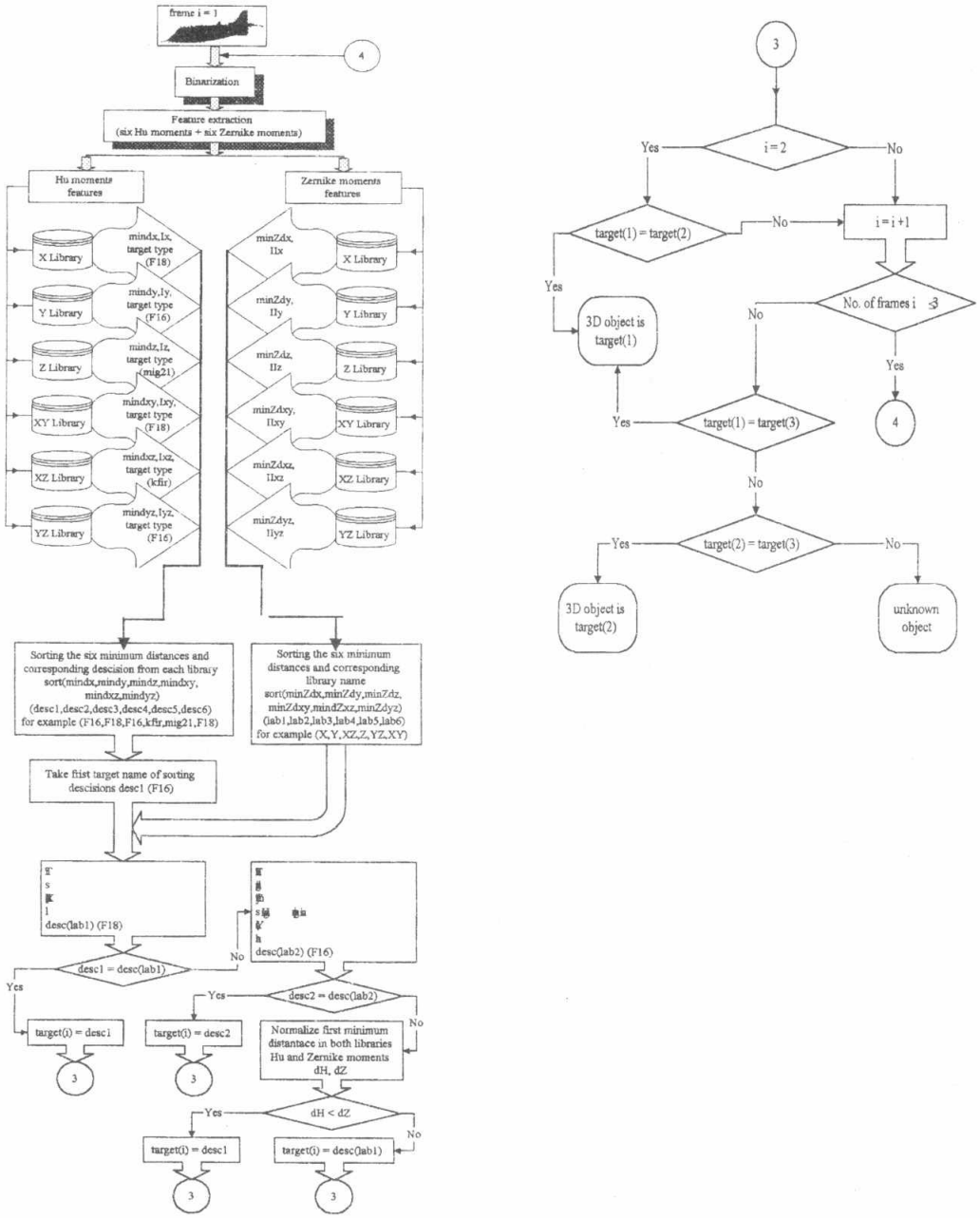
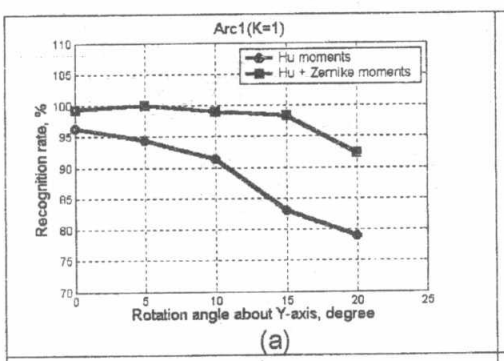
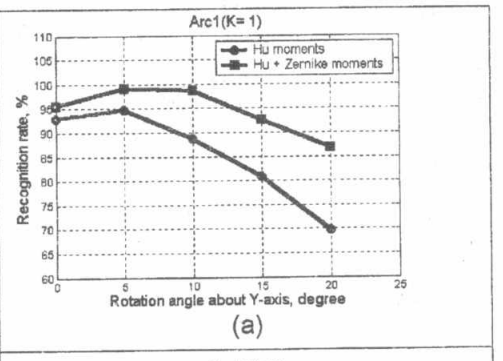
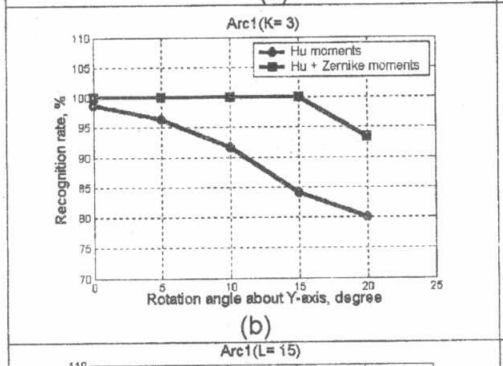
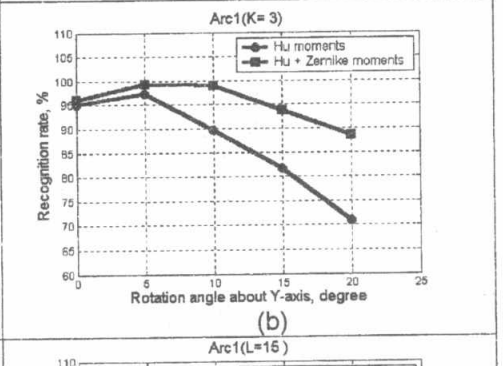


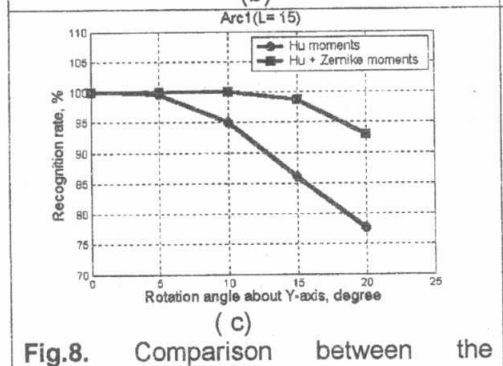
Fig. 5. Functional block diagram of the recognition phase of Hu-Zernike model











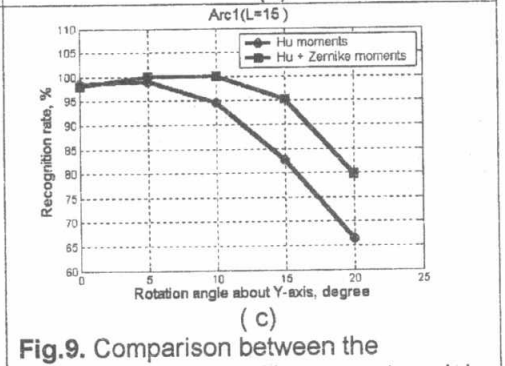


Fig.8. Comparison between the proposed model and Hu-moment model for the five trajectories described in Table (1) using three objects. a) using one 2D view (K=1). b) using three successive 2D views (K=3) c) using three 2D views with frame step L=15.

Fig.9. Comparison between the proposed model and Hu-moment model for the five trajectories described in Table (1) using six objects. a) using one 2D view (K=1). b) using three successive 2D views (K=3) c) using three 2D views with frame step L=15

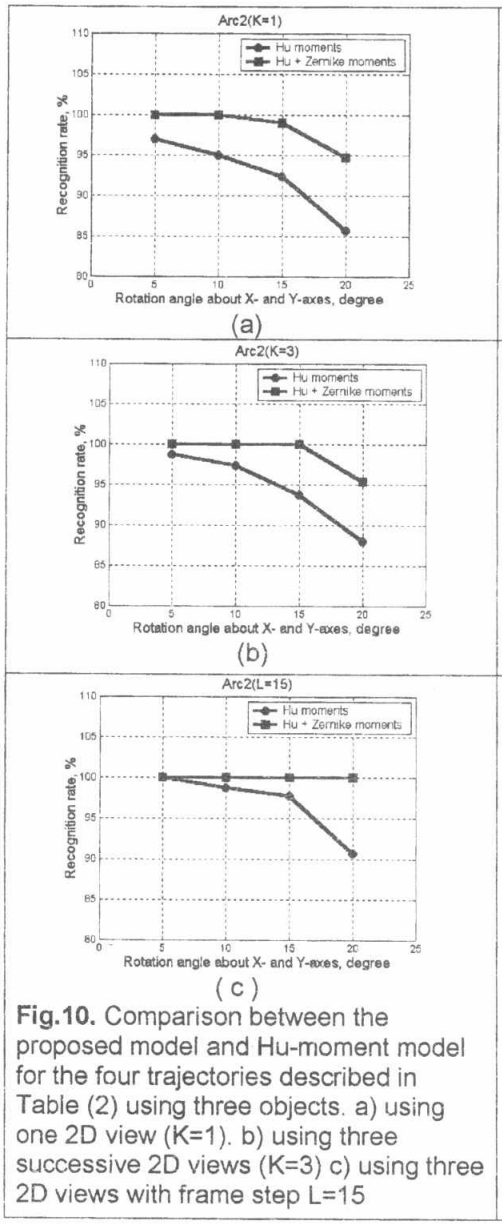


Fig.10. Comparison between the proposed model and Hu-moment model for the four trajectories described in Table (2) using three objects. a) using one 2D view (K=1). b) using three successive 2D views (K=3) c) using three 2D views with frame step L=15

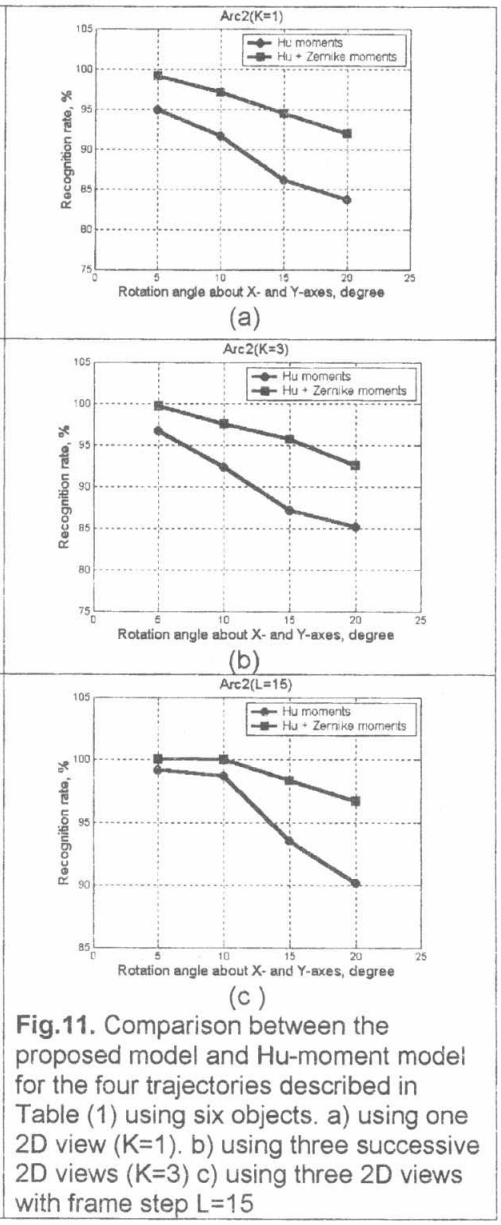


Fig.11. Comparison between the proposed model and Hu-moment model for the four trajectories described in Table (1) using six objects. a) using one 2D view (K=1). b) using three successive 2D views (K=3) c) using three 2D views with frame step L=15

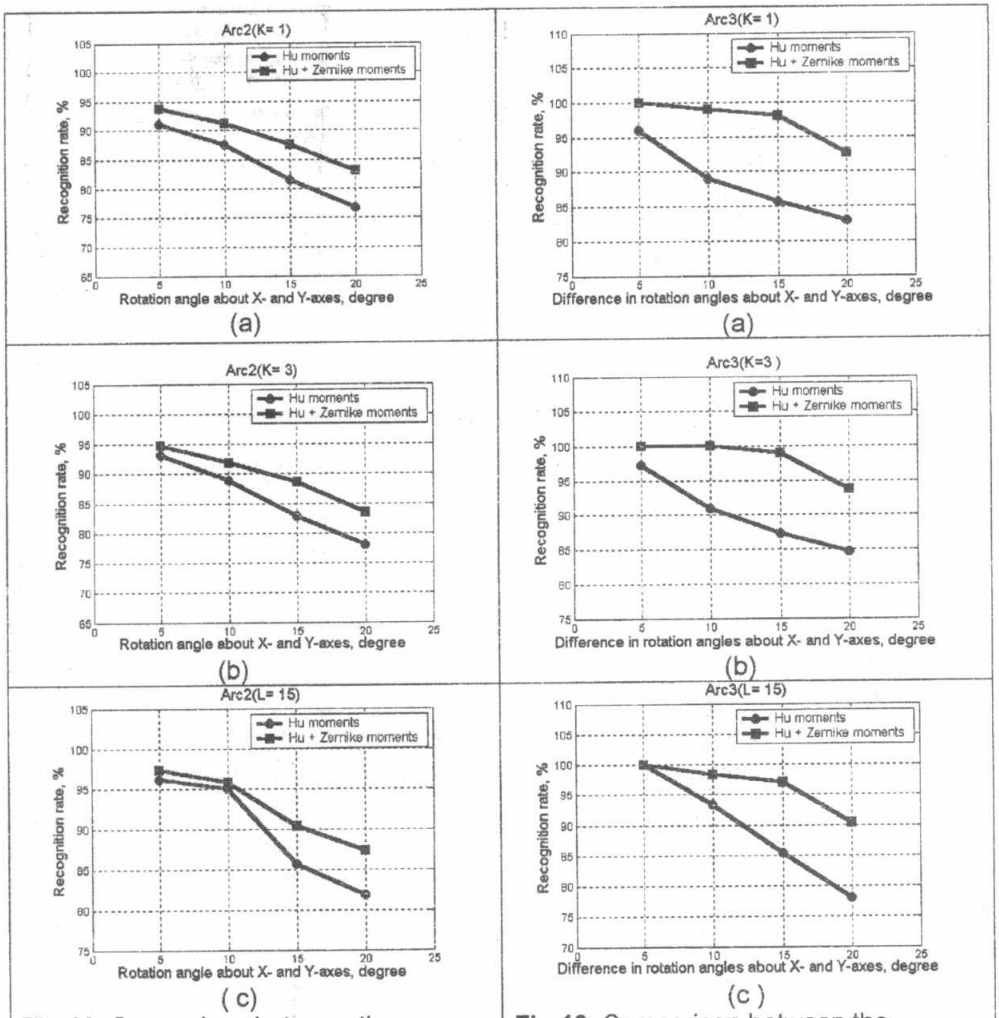
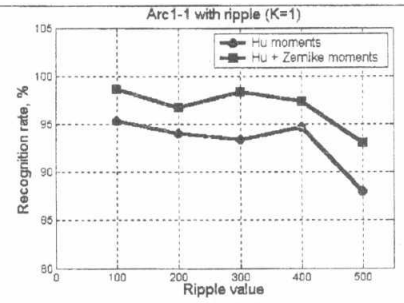
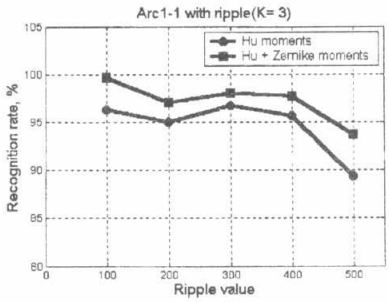


Fig.12. Comparison between the proposed model and Hu-moment model for the five trajectories described in Table (2) using nine objects. a) using one 2D view (K=1). b) using three successive 2D views (K=3) c) using three 2D views with frame step L=15

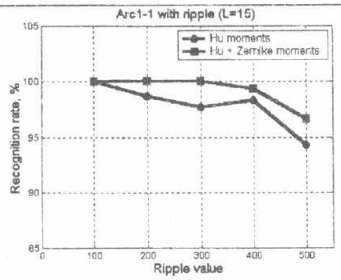
Fig.13. Comparison between the proposed model and Hu-moment model for the five trajectories described in Table (3) using three objects. a) using one 2D view (K=1). b) using three successive 2D views (K=3) c) using three 2D views with frame step L=15



(a)



(b)



(c)

Fig.14. Comparison between the proposed model and Hu-moment model for the arc1-1 affected by ripple noise a) using three objects. b) using six objects. c) using nine objects.

Trajectory	Aircraft pose	
	Frame No. 1	Frame No. 150
Arc1_1	X= 0° ;	X= 0° ;Y= 0° ;
Arc1_2	X= 0° ;	X= 0° ; Y= 5° ;
Arc1_3	X= 0° ;	X= 0° ;Y= 10°
Arc1_4	X= 0° ;	X= 0° ;Y= 15°
Arc1_5	X= 0° ;	X= 0° ;Y= 20°

Table 1. First trajectory set

Trajectory	Aircraft pose	
	Frame No. 1	Frame No. 150
Arc2_1	X= 0° ; Y=	X= 5° ; Y= 5°
Arc2_2	X= 0° ; Y=	X= 10° ; Y=
Arc2_3	X= 0° ; Y=	X= 15° ; Y=
Arc2_4	X= 0° ; Y=	X=20° ; Y=

Table 2. Second trajectory set

Trajectory	Aircraft pose	
	Frame No. 1	Frame No. 150
Arc3_1	X= 0° ;	X= 5° ; Y=
Arc3_2	X= 0° ;	X= 20° ; Y=
Arc3_3	X= 0° ;	X= 5° ; Y= 20°
Arc3_4	X= 0° ;	X=10° ; Y=

Table 3. Third trajectory set