Feature based 3D Object Recognition using Artificial Neural Networks

Abdel karim Baareh

Computer Science Department,
Al-Balqa Applied University,
Ajloun College,
Amman, Jordan

Alaa F. Sheta

Computer Science Department,
The World Islamic Science and
Education (WISE) University,
Amman, Jordan

Mohammad S. Al-Batah

Computer Science Department
Faculty of Sciences and
Information Technology, Jadara
University, Jordan

ABSTRACT

The recognition of objects is one of the main goals for computer vision research. This paper formulates and solves the problem of three-dimensional (3D) object recognition for Polyhedral objects. A multiple view of 2D intensity images are taken from multiple cameras and used to model the 3D objects. The proposed methodology is based on extracting set of features from the 2D images which include the Affine, Zernike and Hu moments invariants to be used as inputs to train artificial neural network (ANN). Various architectures of ANN were explored to recognize a shape of Polyhedral objects. The experiments results show that 3D objects can be sufficiently modeled and recognized by set of multiple 2D views. The best ANN architecture was twenty input and single output model.

General Terms

Pattern Recognition, Artificial Neural Networks.

Keywords

3D object recognition, moments, features extraction, classifications, back propagation.

1. INTRODUCTION

Computer vision systems mainly consider the recognition of 3D objects as a challenging process. Many applications were reported such as the automation on the assembly line, inspection of integrated circuit chips to detect defects in them, security in face and fingerprint recognition, medical diagnosis and detection of abnormal cells that may indicate cancer, remote sensing for automated recognition of possible hostile terrain to generate maps and aids for the visually impaired of mechanical guide dogs. A more detailed list of computer vision applications can be found in [1].

Feature-based 3D object recognition techniques are well known in the literature [2]. These types of techniques found to be successful especially for well structured objects. This means that objects which have unique features or characteristics such as edges can be well recognized. These techniques work, by collecting a set of 2D images for a certain objects (i.e. seen) from various angles and use it for matching purposes. Feature extraction algorithms are used to select the best set of features which can uniquely distinguish the object from the collected 2D images. In fact, pattern recognition techniques normally use gathered information from 2D captured images for an object and do not take in consideration the 3D geometric during the matching process [3, 4].

Many soft computing techniques such as genetic algorithms (GAs) were explored to solve the 3D object recognition problem. For example, in [5], authors investigated the application of genetic algorithms for recognizing 3D objects

from two-dimensional intensity images, assuming orthographic projection. Authors used a variety of 2D views depicting a 3D object under the case of orthographic projection which was expressed as a linear combination of a small number of 2D views of an object. In [6], authors developed a mathematical models based Takagi Sugano fuzzy membership models to describe the relationship between set features extracted from 2D images of a 3D object and a model class. The mathematical models helped in understanding the characteristics of the model and the advantages of fuzzy modeling in solving such a problem.

Fig 1: Polyhedral objects

In this paper, we introduce our initial idea in developing ANN model which counts on a set of features extracted from 2D images to recognize a set of 3D objects. Figure 1 shows the eleven objects to be recognized using the proposed ANN. Polyhedral objects have been used in order to analyze the performance of the proposed ANN system. Polyhedral objects, under study, consist of eleven 3D objects. The test set contains object of simple 3D shape like box, cylinder, sphere, trapezoid etc.

2. BACKPROPAGATION LEARNING ALGORITHM

Recently, ANN have been used successfully to solve a variety of problems such as finance, cancer detection [7], science, forecasting [8], feature extraction [9], classifications [10, 11], face recognition [12], Fingerprint recognition [13] etc. In this paper, we use a feed forward artificial neural network to solve the 3D object recognition problem based a set of collected 2D images. Feed forward ANN normally consists of three types of layers. These layers are the input, hidden and output layers, respectively. A weighted sum of the neuron inputs specifies the activation (i.e. sigmoid) function argument. This activation function is assumed to be nonlinear [14]. Let $x_1(p)$, $x_2(p)$, ..., $x_n(p)$ be the network inputs, and let $y_1(p)$, $y_2(p)$, ..., $y_n(p)$ be the desired output. p is defined as the iteration number.

The back propagation neural network function can be expressed as [15]:

1. Calculating the output from the hidden layer as shown in Equation 1.

$$y_j(p) = \text{sigmoid}[\sum_{i=1}^n x_i(p)w_{ij}(p) - \theta_j] \quad (1)$$

where n is the number of inputs of neuron j in the hidden layer number, w_{ij} are the weights between the input layer and the hidden layer and between the hidden layer and the output layer, θ is a threshold value. The sigmoid function adopted in this study is given in Equation 2.

$$y_j(p) = \frac{1}{1 + e^{-x_j(p)}} \quad (2)$$

2. Calculating the output from the output layer as shown in Equation 3.

$$y_k(p) = \text{sigmoid}[\sum_{j=1}^m x_{jk}(p)w_{jk}(p) - \theta_k] \quad (3)$$

where m is the number of inputs of neuron k in the output layer

3. Calculating the Error Gradient from the output layer as seen in Equation 4.

$$\delta_k(p) = y_k(p)[1 - y_k(p)]e_k(p) \quad (4)$$

where $e_k(p)$ is the error at the output layer.

$$e_k(p) = y_{d+k}(p) - y_k(p) \quad (5)$$

4. The ANN weights can be computed as given in Equation 6.

$$\Delta w_{jk}(p) = \alpha y_j(p) + \delta_k(p) \quad (6)$$

To readjust the weights of the ANN we use Equation 7.

$$w_{jk}(p+1) = w_{jk}(p) + \Delta w_{jk}(p) \quad (7)$$

To calculate the gradient error in the hidden layer we use Equation 8.

$$\delta_j(p) = y_j(p)[1 - y_j(p)] \sum_{k=1}^i \delta_k(p)w_{jk}(p) \quad (8)$$

5. Calculating the weights again from Equation 9.

$$\Delta w_{ij}(p) = \alpha x_i(p) + \delta_j(p) \quad (9)$$

Again we readjust the weights as given in Equation 10.

$$w_{ij}(p+1) = w_{ij}(p) + \Delta w_{ij}(p) \quad (10)$$

3. CAMERA OBJECT SETUP

In this section, a methodology for camera object setup is presented. Each selected object is positioned in its stable condition at the centre of the turntable. The turntable is a circular horizontal platform that can be rotated 360 degree. Illumination using controlled lighting condition is provided to get the image of an object. This is how we can avoid shadow and reflection on the images. Three CCD cameras were used to obtain a number of 2D images of the 3D objects. Camera 1 was fixed at the top of the object, while cameras 2 and 3 were fixed at 45° from perpendicular view. The camera-object setup is shown in Figure 2. Since all cameras have the same

distance from the centre of the turntable, all cameras must have the same focal lengths.

Typically, the height of the video camera is adjusted to be within 1500 mm from the object to obtain a sufficiently large measurement area. However, the camera can be placed at different positions from the object without effecting the image's feature extraction as the moments features are invariant to object rotation, translation and size scaling in the image. The lens is adjusted accordingly to obtain a clear and sharp image for analysis. The maximum resolution of the input image is fixed to 1024 × 776. After an object of interest is placed at the centre of the turntable, the 2D images of the object are captured and stored in a lossless digital format, bitmap (bmp), in the computer memory. Each rotation, the object will rotate 50° and so on until 360° is completed. Hence, for each object, 72 image sets are obtained.

Fig 2: Hardware setup for image acquisition [16]

4. FEATURE EXTRACTION

Moment is considered significant features for object recognition. Each object can be recognized if carefully selected features are chosen [17-19]. 2D moments are simple mathematical features to be computed [20]. The object surface area, position, orientation can describe the main characteristics of an object [21]. Various types of moments were presented in the literature. They include Hu moments [22], Zernike moments [23], and Affine moments [24] which were used in many object recognition applications. In this paper, we adopted seven Hu [22], eleven Zernike [23], and six Affine moments [24] features to recognize the 3D objects.

4.1 Hu Moment Invariants

In this study, we used the geometrical moments for feature extraction in 3D object shape classification. The invariance property of Hu, Zernike and Affine moments against geometrical transformations like scaling, translation and rotation makes it a good candidate of feature extractor to be used for object recognition. The ease of the 2D moment computation helps reducing the processing time and increases the system effectiveness and for real time computer simulation. In order to understand how to utilize moment invariant method, let f_b be a binary digital image with size $M \times N$ (i.e. where M and N are the image width and height, respectively) and $f_b(x,y)$ is the grey level value for the pixel at row x and column y . The two-dimensional moments of order $(p+q)$ of $f_b(x, y)$ which is variant to the scale, translation and rotation is defined as

$$m_{pq} = \sum_{x=1}^M \sum_{y=1}^N x^p y^q f_b(x, y) \quad (11)$$

The central moments of order $(p + q)$ of $f_b(x, y)$ is defined as

$$\mu_{pq} = \sum_{x=1}^M \sum_{y=1}^N (x - x_c)^p (y - y_c)^q f_b(x, y) \quad (12)$$

where x_c and y_c are the centre of mass of the object defined as:

$$x_c = \frac{m_{10}}{m_{00}} \quad y_c = \frac{m_{01}}{m_{00}} \quad (13)$$

The moment invariant under scale is defined as

$$\eta_{pq} = \frac{\mu_{pq}'}{(\mu_{00}')^\gamma} \quad (14)$$

where

$$\gamma = \frac{p+q}{2} + 1 \quad (15)$$

and

$$\mu_{pq}' = \frac{\mu_{pq}}{\alpha^{(p+q+2)}} \quad (16)$$

Normalized un-scaled central moment is then given by

$$\mathcal{G}_{pq} = \frac{\mu_{pq}}{(\mu_{00})^\gamma} \quad (17)$$

From the second and third order moments, a set of seven invariant moments which are invariant to translation, rotation and scale derived by Hu [22] are:

$$\varphi_1 = \mathcal{G}_{20} + \mathcal{G}_{02} \quad (18)$$

$$\varphi_2 = (\mathcal{G}_{20} - \mathcal{G}_{02})^2 + 4\mathcal{G}_{11}^2 \quad (19)$$

$$\varphi_3 = (\mathcal{G}_{30} - 3\mathcal{G}_{12})^2 + (3\mathcal{G}_{21} - \mathcal{G}_{03})^2 \quad (20)$$

$$\varphi_4 = (\mathcal{G}_{30} + \mathcal{G}_{12})^2 + (\mathcal{G}_{21} + \mathcal{G}_{03})^2 \quad (21)$$

$$\begin{aligned} \varphi_5 = & (\mathcal{G}_{30} - 3\mathcal{G}_{12})(\mathcal{G}_{30} + \mathcal{G}_{12}) \\ & [(\mathcal{G}_{30} + \mathcal{G}_{12})^2 - 3(\mathcal{G}_{21} - \mathcal{G}_{03})^2] \\ & + (3\mathcal{G}_{21} - \mathcal{G}_{03})(\mathcal{G}_{21} + \mathcal{G}_{03}) \\ & [3(\mathcal{G}_{30} + \mathcal{G}_{12})^2 - (\mathcal{G}_{21} + \mathcal{G}_{03})^2] \end{aligned} \quad (22)$$

$$\begin{aligned} \varphi_6 = & (\mathcal{G}_{20} - \mathcal{G}_{02})[(\mathcal{G}_{30} + \mathcal{G}_{12})^2 - (\mathcal{G}_{21} + \mathcal{G}_{03})^2] \\ & + 4\mathcal{G}_{11}(\mathcal{G}_{30} + \mathcal{G}_{12})(\mathcal{G}_{21} + \mathcal{G}_{03}) \end{aligned} \quad (23)$$

$$\begin{aligned} \varphi_7 = & (3\mathcal{G}_{21} - \mathcal{G}_{03})(\mathcal{G}_{30} + \mathcal{G}_{12})[(\mathcal{G}_{30} + \mathcal{G}_{12})^2 \\ & - 3(\mathcal{G}_{21} - \mathcal{G}_{03})^2] - (\mathcal{G}_{30} - 3\mathcal{G}_{12})(\mathcal{G}_{21} + \mathcal{G}_{03}) \\ & [3(\mathcal{G}_{30} + \mathcal{G}_{12}) - (\mathcal{G}_{21} + \mathcal{G}_{03})^2] \end{aligned} \quad (24)$$

where $\mathcal{G}_{pq}$ was obtained from Equation 17.

The algorithm of Hu moment invariants is implemented in [22]:

1. Calculating the value of x_c and y_c based on Equation 13.
2. Computing each value of μ_{00} , μ_{11} , μ_{02} , μ_{20} , μ_{21} , μ_{12} , μ_{03} , and μ_{30} , using Equation 12.
3. Computing each value of $\mathcal{G}_{11}$, $\mathcal{G}_{02}$, $\mathcal{G}_{20}$, $\mathcal{G}_{21}$, $\mathcal{G}_{12}$, $\mathcal{G}_{03}$, and $\mathcal{G}_{30}$, using Equation 17.
4. Calculating the value of Hu moments φ_1 – φ_7 according to Equations 18 until 24.

4.2 Zernike Moments

Zernike moments are defined over a set of complex polynomials which is a complete orthogonal set over the unit disk $x^2 + y^2 \leq 1$ and can be denoted as in Equation 25.

$$A_{mn} = \frac{m+1}{\pi} \int \int_{x^2+y^2 \leq 1} f(x, y) [V_{mn}(x, y)]^* dx dy \quad (25)$$

where $m = 0, 1, 2, \dots, \infty$. $f(x, y)$ represents the pixel value of the image at x and y coordinates, $V_{mn}(x, y)$ is the Zernike polynomial, complex conjugate is represented by $*$ symbol and n must satisfy two conditions:

1. $m - |n| = \text{even}$, and
2. $|n| \leq m$.

Zernike polynomial $V_{mn}(x, y)$ can be expressed in polar coordinate using the relation

$$V_{mn}(r, \theta) = R_{mn}(r) \exp(jn\theta) \quad (26)$$

where (r, θ) denoting as disk unit and $R_{mn}(r)$ is orthogonal radial polynomial given by

$$R_{mn}(r) = \sum_{s=0}^{\frac{m-|n|}{2}} (-1)^s F(m, n, s, r) \quad (27)$$

where

$$F(m, n, s, r) = \frac{(m-s)!}{s! \left(\frac{m+|n|}{2} - s\right)! \left(\frac{m-|n|}{2} - s\right)!} r^{m-2s} \quad (28)$$

for a digital image $f_b(x, y)$, Expression (25) can be denoted with

$$A_{mn} = \frac{m+1}{\pi} \sum_x \sum_y f_b(x, y) [V_{mn}(x, y)]^* \quad (29)$$

Pejnovic et al. [23] derived Zernike moments in terms of central moments as shown in Equation 12. Up to the 4-th order $(p + q < 4)$, the Zernike moments are given by:

$$S_1 = A_{20} = \frac{3(2(\mu_{20} + \mu_{02}) - 1)}{\pi} \quad (30)$$

$$S_2 = |A_{22}|^2 = \frac{9((\mu_{20} - \mu_{02})^2 + 4(\mu_{11})^2)}{\pi^2} \quad (31)$$

$$S_3 = |A_{33}|^2 = \frac{16((\mu_{03} - 3\mu_{21})^2 + (\mu_{30} - 3\mu_{12})^2)}{\pi^2} \quad (32)$$

$$S_4 = |A_{31}|^2 = \frac{144((\mu_{03} + \mu_{21})^2 + (\mu_{30} + \mu_{12})^2)}{\pi^2} \quad (33)$$

$$S_5 = \frac{A_{33}^* A_{31}^3 + A_{33}(A_{31}^*)^3}{\pi^4} ((\mu_{03} - 3\mu_{21})(\mu_{03} + \mu_{21})(\mu_{03} - \mu_{21})^2 - 3(\mu_{30} + \mu_{12})^2 - (\mu_{30} + 3\mu_{12})(\mu_{30} + \mu_{12}) + (\mu_{30} + \mu_{12})^2 - 3(\mu_{03} + \mu_{21})^2) \quad (34)$$

$$S_6 = A_{31}^2 A_{22}^* + (A_{31}^*)^2 A_{22} = \frac{864}{\pi^3} ((\mu_{02} - \mu_{20})(\mu_{03} + \mu_{21})^2 (\mu_{30} - \mu_{12})^2 + 4\mu_{11}(\mu_{03} + \mu_{21})^2 (\mu_{30} + \mu_{12})) \quad (35)$$

$$S_7 = |A_{44}|^2 = \frac{25}{\pi^2} ((\mu_{40} - 6\mu_{22} + \mu_{04})^2 + 16(\mu_{31} - \mu_{13})^2) \quad (36)$$

$$S_8 = |A_{42}|^2 = \frac{25}{\pi^2} (4(\mu_{04} - \mu_{40}) + 3(\mu_{20} - \mu_{02})^2 + 4(4(\mu_{31} + \mu_{13}) - 3(\mu_{11})^2)) \quad (37)$$

$$S_9 = A_{40} = \frac{5}{\pi} (6(\mu_{40} + 2\mu_{22} + \mu_{04}) - 6(\mu_{20} + \mu_{02}) + 1) \quad (38)$$

$$S_{10} = A_{44}^* A_{42}^2 + A_{44}(A_{42}^*)^2 = \frac{250}{\pi^3} ((\mu_{40} - 6\mu_{22} + \mu_{04})(4(\mu_{04} - \mu_{40}) + 3(\mu_{20} + \mu_{02})^2) - 4(4(\mu_{31} + \mu_{13}) - 3\mu_{11})^2) \quad (39)$$

$$S_{11} = A_{42}^* A_{22} + A_{42}^* A_{22} = \frac{30}{\pi^2} (4(\mu_{04}\mu_{40}) + 3(\mu_{20} - \mu_{02})(\mu_{02} - \mu_{20}) + 4\mu_{11}(4(\mu_{31} + \mu_{13}) - 3\mu_{11})) \quad (40)$$

The following conditions were adopted for calculating the Zernike moments as presented in [23]:

1. Calculating the value of x_c and y_c based on Equation 13.
2. Computing each value of μ_{00} , μ_{11} , μ_{02} , μ_{20} , μ_{21} , μ_{12} , μ_{22} , μ_{03} , μ_{30} , μ_{31} , μ_{13} , μ_{40} , and μ_{04} using Equation 12.
3. Calculating the value of Zernike moments $S_I - S_{11}$ according to Equations 30 until 40.

4.3 Affine Moments

The six Affine adopted moment invariants features presented in [24] are defined as follows:

$$I_1 = \frac{1}{\mu_{00}^4} (\mu_{20}\mu_{02} - \mu_{11}^2) \quad (41)$$

$$I_2 = \frac{1}{\mu_{00}^{10}} (\mu_{30}^2\mu_{03}^2 - 6\mu_{30}\mu_{21}\mu_{12}\mu_{03} + 4\mu_{30}\mu_{12}^3 + 4\mu_{03}\mu_{21}^3 - 3\mu_{21}^2\mu_{12}^2) \quad (42)$$

$$I_3 = \frac{1}{\mu_{00}^7} (\mu_{20}(\mu_{21}\mu_{03} - \mu_{12}^2) - \mu_{11}(\mu_{30}\mu_{03} - \mu_{21}\mu_{12}) + \mu_{02}(\mu_{30}\mu_{12} - \mu_{21}^2)) \quad (43)$$

$$I_4 = \frac{1}{\mu_{00}^{11}} (\mu_{20}^3\mu_{03}^2 - 6\mu_{20}^2\mu_{11}\mu_{12}\mu_{03} - 6\mu_{20}^2\mu_{21}\mu_{02}\mu_{03} + 9\mu_{20}^2\mu_{02}\mu_{12}^2 + 12\mu_{20}\mu_{11}\mu_{03}\mu_{21} + 6\mu_{20}\mu_{11}\mu_{02}\mu_{30}\mu_{03} - 18\mu_{20}\mu_{11}\mu_{02}\mu_{21}\mu_{12} - 8\mu_{11}^3\mu_{03}\mu_{30} - 6\mu_{20}\mu_{02}^2\mu_{30}\mu_{12} + 9\mu_{20}\mu_{02}^2\mu_{21}^2 + 12\mu_{11}^2\mu_{02}\mu_{30}\mu_{12} - 6\mu_{11}^2\mu_{02}^2\mu_{30}\mu_{21} + \mu_{02}^3\mu_{30}^2) \quad (44)$$

$$I_5 = \frac{1}{\mu_{00}^6} (\mu_{40}\mu_{04} - 4\mu_{31}\mu_{13} + 3\mu_{22}^2) \quad (45)$$

$$I_6 = \frac{1}{\mu_{00}^9} (\mu_{40}\mu_{04}\mu_{22} + 2\mu_{31}\mu_{22}\mu_{13} - \mu_{40}\mu_{13}^2 - \mu_{04}\mu_{31}^2 + \mu_{22}^3) \quad (46)$$

The algorithm of Affine moments is implemented as given in [24] and summarized as follows:

1. Calculating the value of x_c and y_c based on Equation. 13.
2. Computing each value of μ_{00} , μ_{11} , μ_{02} , μ_{20} , μ_{21} , μ_{12} , μ_{22} , μ_{03} , μ_{30} , μ_{31} , μ_{13} , μ_{40} , and μ_{04} using Equation 12.
3. Calculating the value of Affine moments ($I_I - I_6$) according to Equations 41 until 46.

In this paper, 20 moment features are extracted. The features include 7 Hu moments ($Q_I - Q_7$), 7 Zernike moments ($S_I - S_7$), and 6 Affine moments ($I_I - I_6$). After the feature extraction process, the 20 features are then fed as input data to the feed forward artificial neural network for the classification stage. Figure 3 and 4 show the collected features measurements which will be used for ANN training purposes. We had 396 examples for training and the same number of examples for testing. The 20 inputs are shown with many variations.

Fig 3: A set of first ten features used in the learning process (f1 to f10)

Fig 4: A set of second ten features used in the learning process (f11 to f20)

5. ANN CLASSIFICATIONS MODEL

Our problem is to classify the difference between eleven objects. The collected dataset consists of a total of 792 images which were captured by 3 cameras. These images are separated into two groups. Group 1 has 396 images for training data and group 2 has 396 images for testing data. Images with the camera angles at $0^\circ, 10^\circ, 20^\circ \dots 350^\circ$ condition are considered for training, and the rest of the images with angles $5^\circ, 15^\circ, 25^\circ \dots 355^\circ$ condition were considered for testing.

5.1 Neural Network Structure

The architecture of the network used for the classification of the polyhedral object is a multi-layer feed forward network. It consists of an input layer, one hidden layer, and an output layer. The input layer contains a number of neuron corresponding to the set of features used which is 20 inputs in our case. The hidden layer consists of nonlinear hidden units and performs nonlinear transformation which is selected to have a variety of neurons. In our case, we tried 10, 15, 20, 25 and 30 neurons in the hidden layer. The hidden units are fully connected to both the input and output. The nonlinearity is provided by the activation function of the hidden units. The activation function was selected as in a sigmoid form. The output layer consists of one output neuron producing the corresponding object classification type. In our case we used 11, 4 and one output. The output layer node has a linear activation function.

5.2 Optimal Neurons in the Hidden Layer

To select the optimal number of neuron in the hidden layer a number of experiments was completed. The network was trained using the BP algorithm with various numbers of neurons in the hidden layer starting with 10 neurons up to 30 neurons, with step 5. We found that the optimal number of neurons in the hidden layer, to achieve minimum error, which found to be 10 neurons with the 11 and 4 outputs and 15 neurons with single output neural networks. The NN developed models are shown in Figures 5, 6, and 7.

5.3 Performance Evaluation

All of the proposed network architecture was able to produce the best classification results in both training and testing cases with small number of epochs. We calculated the computed variance-account-for (VAF) for evaluation purposes.

Fig 5: ANN for 20 inputs and 11 outputs

Fig 6: ANN for 20 inputs and 4 outputs

Fig 7: ANN for 20 inputs and 1 output

$$VAF = \left[1 - \frac{\text{var}(y)}{\text{var}(g)} \right] \times 100\% \quad (47)$$

where y represents the actual classification values and g is the computed classification values for each output. The computed VAF was 100% in the training cases and almost the same in the testing case for a single output ANN. The neural network back propagation learning algorithm achieved the required goal and was able to minimize the error difference between the actual and estimated response as required. The first two cases, with eleven output and four outputs, produced over 99% recognition rate. The number of neurons in the hidden layers was only ten neurons although in the case of a single output ANN we had to use more neurons in the hidden layers to achieve a better recognition rate (i.e. fifteen neurons).

Fig 8: NNs convergence for 20 inputs and 11 outputs

Figure 8, 9, and 10 show the convergence curves for the three cases we studied. Various numbers of neurons in the hidden layers were explored to find the optimal set of neurons which can be used for solving the recognition problem. The best found in case one and two were ten and in case three were fifteen as stated before. The actual and estimated classes in both training and testing cases for case 3 (i.e. the ANN with single output) is presented in Figure 11 and 12. The VAF performance evaluation criteria were computed in the two cases. It was found as 100% in the training case and 99.9426% in the testing case. This shows the significant of the proposed ANN models.

Fig 9: NNs convergence for 20 inputs and 4 outputs

6. CONCLUSION

In this paper, we explored the use of feed forward ANN to solve the 3D object recognition problem. A multiple view of 2D intensity images are taken from multiple cameras and used to model the 3D objects. ANN was used for matching and classification of the 3D objects. A set of features which include the Affine, Zernike and Hu moments invariants are selected to train the ANN. Various ANN structures were explored to achieve the recognition of a Polyhedral objects. The experiments show that 3D objects can be modeled and represented by a set of multiple 2D views. In addition, it does not require complex feature sets for 3D object modeling, thus improve processing time for feature extraction stage. We plan

to explore the use of other soft computing techniques to solve this problem such as genetic algorithms.

Fig 10: NNs convergence for 20 inputs and 1 output

Fig 11: Training Samples along with identified and estimated classes with a VAF= 100%

Fig 12: Testing Samples along with identified and estimated classes with a VAF= 99.9426

7. ACKNOWLEDGMENTS

Dr. Mohammad Al-Batah would like to thank Eng. M. Khusairi Osman from University Sains Malaysia, for helping in the 3D data collection. Prof. Sheta is on leave from the Electronics Research Institute (ERI), Cairo, Egypt.

8. REFERENCES

- [1] Szeliski, R., 2010, *Computer Vision: Algorithms and Applications*, Springer, New York.
- [2] Sharath, P., Chitra, D., and Anil, K., R., 1993, Feature Detection for 3D Object Recognition and Matching, In *Proceedings, of SPIE Conference on Geometric Methods In Computer Vision II*, pp. 366-377.
- [3] Murase, H., and S. K. Nayar, "Visual Learning and Recognition of 3-D Objects from Appearance", *International Journal of Computer Vision* 14, 1995, pp. 5-24.
- [4] Selinger, A. and R. Nelson, "A Perceptual grouping Hierarchy for Appearance-Based 3D Object Recognition", *Computer Vision and Image Understanding* Vol.76, No.1, 1999, pp. 83-92.
- [5] Bebis, G, Louis, S, and Fadali, S, 1998, Using Genetic Algorithms for 3-D Object Recognition, In *Proceeding 11th Int. Conf. Computer Applications in Industry and Engineering*, Las Vegas, NV, pp. 13-16.
- [6] Sheta, A., Baareh, A. and Al-Batah, M., "3D Object Recognition Using Fuzzy Mathematical Modeling of 2D Images", Accepted for publication at the 3rd International Conference on Multimedia Computing and Systems, May 10-12, Tangier, Morocco, 2012.
- [7] Braik, M., Sheta, A., "A New Approach for Potentially Breast Cancer Detection Using Extracted Features and Artificial Neural Networks", *Journal of Intelligent Computing* Vol. 2, No. 2, June 2011, pp.54-71.
- [8] Baareh, A. K., Sheta, A., AL Khnaifes K., "Forecasting River Flow in the USA: A Comparison between Auto-Regression and Neural Network Non-Parametric Models", *Journal of Computer Science*, Vol. 2, No.10, 2006, pp. 775-780.
- [9] Al-Batah, M. S., Mat Isa, N. A., Zamli, K. Z., and Azizli, K. A, Modified Recursive Least Squares algorithm to train the Hybrid Multilayered Perceptron (HMLP) network, *Applied Soft Computing*, Vol. 10, No. 1, 2010, pp. 236-244.
- [10] Seethe, M., Muralikrishna, I.V., Deekshatulu. B. L., *Artificial Neural Networks and Other Methods of Image Classifications*, *Journal of theoretical and applied information technology*, 2005, pp. 1039-1053.
- [11] Honjun Lu., Rudy S., huan lui., 1996, Effective data mining Using Neural Networks, *IEEE Transactions On Knowledge and Data Engineering*, Vol. 8, No. 6, , pp. 957-961.
- [12] Radha, V., Nallammal., N, October 19-21, 2011, "Neural Network Based Face Recognition Using RBFN Classifier" In *Proceedings of the World Congress on Engineering and Computer Science*, Vol. I, San Francisco, USA.
- [13] Darwis, Y. and, Sheta, A., 2008, "Minutiae Extraction for Fingerprint Recognition" In *Proceedings of the Fifth IEEE International Multi-Conference on System, Signal and Devices (SSD'08)*, Amman, Jordan.
- [14] Baareh, A., Sheta, A, AL Khnaifes K. 2006, "Forecasting river flow in the USA: a comparison between auto-regression and neural network non-parametric models", *SMO'06 Proceedings of the 6th WSEAS International Conference on Simulation, Modeling and Optimization*, pp. 7-12.
- [15] Michael, N., 2005 *Artificial Intelligence: A Guide to Intelligent Systems*, Pearson Education Limited.
- [16] Mat Isa., Zamli, K. Z., and Al-Batah, M. S., "Automated intelligent real-time system for aggregate classification", *International Journal of Mineral Processing*, Vol. 100, No. 1-2, pp. 41 – 50, 2011.
- [17] Munoz-Rodriguez, J. A., Asundi, A., and Rodriguez-Vera, R, "Recognition of a Light Line Pattern by Hu Moments for 3-D Reconstruction of a Rotated Object", *Journal of Optics and Laser Technology*, Vol. 37, No. 2, 2005, pp. 131-138.
- [18] Realpe, A., and Velazquez, C, "Pattern Recognition for Characterization of Pharmaceutical Powders", *Journal of Powder Technology*, Vol. 169, No. 2, 2006, pp. 108-113.
- [19] Rizon, M., Yazid, H., Saad, P., Md-Shakaff, A. Y., Saad, A. R., Mamat, M. R., Yaacob, S., Desa, H., and Karthigayan, M, "Object Detection Using Geometric Invariant Moment, *American Journal of Applied Sciences*", Vol. 2, No. 6, 2006, pp. 1876-1878.
- [20] Mat Isa, N. A., Al-Batah, M. S., Zamli, K. Z., Azizli, K. A., Joret, A., and Mat Noor, N. R., "Suitable features selection for the HMLP and MLP networks to identify the shape of aggregate", *Construction and Building Materials*, Vol. 22, No. 3, 2008, pp. 402-410.
- [21] Al-Batah, M. S., Mat Isa, N. A., Zamli, K. Z., Sani, Z. M., and Azizli, K. A, A novel aggregate classification technique using moment invariants and cascaded multilayered perceptron network, *International Journal of Mineral Processing*, Vol. 92, No. (1-2), 2009, pp. 92-102.
- [22] Hu, M. K. Visual, 1962, *Pattern Recognition by Moment Invariants*, computer methods in image analysis, *IEEE Transactions on Information Theory*, Vol. 8, No. 2, pp. 179-187.
- [23] Pejnovic, P., Buturovic, L., and Stojiljkovic, Z., 1992 *Object Recognition by Invariants*, In *Proceeding of the 11th IAPR International Conference on Pattern Recognition*, Vol.2, pp. 434-437.
- [24] Kadyrov, A., and Petrou, M., 2001, *Object Descriptors Invariant to Affine Distortions*, In *Proceedings of the British Machine Vision Conference*, pp. 391-400.