Recognition of 3D Objects from 2D Images – Some Issues

B. Liu and P. Wang, *IAPR Fellow*
Northeastern University
Boston, MA 02115
bingl@ccs.neu.edu, pwang@ccs.neu.edu
(617)373-3711(O), 373-5121121(Fax)

Abstract

This paper presents a simple methods for visualizing, understanding, interpreting, and recognizing 3D objects from 2D images. It extended the linear combination methods, uses parallel pattern matching and can handle 3D rigid concave objects as well convex objects and articulated objects, yet, needs only a very small number of learning samples. Some real images are illustrated, with future research discussed.

Keywords: 3D object recognition, 2D images, realization, visualization, understanding, recognition, concave objects, convex objects, rigid objects, linear combination, parallel pattern matching, articulated objects, learning

1 Introduction

In general, 3D (3-dimensional) object recognition process involves 2D (2-dimensional) input images comparing with the model images. An image, which is a 2D picture of object, normally only represents one view of the object, and objects usually have different view from different viewing points. Therefore, one of the most difficult and challenging problems in computer vision and recognition is how to recognize a 3D objects from 2D images. Many research works have been done in this field in the past few years [9].

Basri's computational approach [2] uses viewer-centered representations to recognize 3D objects. This scheme requires storing only a small number of views, but can only handle convex objects. Some more details and illustrations can be found in [10].

Baird and Wang's recognition algorithm[1], which uses gradient descent and the universal 3-D array grammar concept, tends to reach the same 3D interpretations of 2D line drawings that humans do. They interpret the line drawings without models by using Marill's minimum Standard Deviation of Angle (MSDA) principle [6], yet is faster and more accurate.

Wang introduced a heuristic parallel approach [8] for 3D line image analysis by using the concept of coordinated graph, layered graph representation and parallel matching techniques. Wang also presented a new hybrid approach [7] for recognizing complicated objects including concave, partially self-occluded and articulated object from 2D line drawings, yet it was only tested on noise-free well behaved polyhedrals.

On the other hand, as we know, an articulated object is a collection of links connected by joints. Each link is a rigid component. It can move independently of the other links when only is joints constrain its motion. The recognition of articulated object is complicated because it involves not only the different views by rotating the whole object but also the views of rotationg part of the object. However, articulated objects are of special interests and importance since they include most of the industrial robots and man-made factory tools, such as robot arms, boxes and pairs of scissors. Also, it is very useful in the military field.

Up to date, very few work was done on articulated object recognition. The first attempt to tackle such problem was Brook's famous ACRONYM system using symbolic reasoning [3]. Grimson et al [5] extended the interpretation tree approach to deal with 2-D objects with rotating subparts.

Goldberg and Lowe [4] extended Lowe's system to deal with 3-D articulated objects.

All the methods mentioned above deal with either the 2D line-drawing images, which are already preprocessed, or some kinds of particular articulated objects. Here I present a method using linear combination approach to recognize 3-D articulated object. The input of the method is the 3D object itself without any preprocessing. It generally applies to many kinds of objects and is simple, straight and does not need explicit 3D descriptions which the conventional methods request.

2 Linear Combination Method

Linear combination method is based on the observation that novel views of objects can be expressed as linear combination of the stored views. It identifies objects by constructing custom-tailored templates from stored two-dimensional image models. The template-construction procedure just adds together weighted coordinate values from corresponding points in the stored two-dimensional image models. Here, a model is a representation in which

- an image consists of a list of feature points observed in the image
- the model consists of several images - minimally three for polyhedra.

An unknown object is matched with a model by comparing the points in an image of the unknown object with a template-like collection of points produced from the model. In general, an unknown object can be arbitrarily rotated, arbitrarily translated and even arbitrarily scaled relative to an arbitrary original position. From the basis graphic knowledge, an arbitrary rotation and translation of an object transforms the coordinate value of any point on that object according to the following equations:

$$X_\theta = r_{xx}(\theta)X + r_{yx}(\theta)Y + r_{zx}(\theta)Z + t_x$$
$$Y_\theta = r_{xy}(\theta)X + r_{yy}(\theta)Y + r_{zy}(\theta)Z + t_y$$
$$Z_\theta = r_{xz}(\theta)X + r_{yz}(\theta)Y + r_{zz}(\theta)Z + t_z$$

where $r_{ij}(\theta)$ (i,j=x,y,z) is the parameter that shows how much the i coordinate of a point, before rotation, contributes to the j coordinate of the same point after rotation,

and t_s(s=x,y,z) is the parameter that is determined by how much the object is translated.

Based on S. Ullman's concept that three images, each showing four corresponding vertexes, are almost enough to determine the vertexes' relative positions, therefore, at lease three model images are needed and these three model images yield the following equations relating models and unknown object coordinate values to unrotated, untranslated coordinate values, x,y,z.

$$
\begin{aligned}
X_{I1} &= r_{xx}(\theta_1)X + r_{yx}(\theta_1)Y + r_{zx}(\theta_1)Z + t_x(\theta_1) \\
X_{I2} &= r_{xx}(\theta_2)X + r_{yx}(\theta_2)Y + r_{zx}(\theta_2)Z + t_x(\theta_2) \\
X_{I3} &= r_{xx}(\theta_3)X + r_{yx}(\theta_3)Y + r_{zx}(\theta_3)Z + t_x(\theta_3) \\
X_{I0} &= r_{xx}(\theta_0)X + r_{yx}(\theta_0)Y + r_{zx}(\theta_0)Z + t_x(\theta_0)
\end{aligned}
$$

These equations can be viewed as four equations in four unknowns, X, Y, Z and X_{I0}, and can be solved to yield X_{I0} in term of X_{I1} X_{I2} X_{I3} and a collection of four constraints,

$$X_{I0} = \alpha_x X_{I1} + \beta_x X_{I2} + \gamma_x X_{I3} + \delta_x$$

where α_x, β_x, γ_x and δ_x are the constraints required for x-coordinate-value prediction, each of which can be expressed in term of r_s and t_s. In order to determine the constraints value, a few corresponding points are needed, here there are four constraints, therefore, four equations are needed, furthermore, four feature points are required in every image. The four equation are described as follows:

$$
\begin{aligned}
X_{P_1 I_0} &= \alpha_x X_{P_1 I_1} + \beta_x X_{P_1 I_2} + \gamma_x X_{P_1 I_3} + \delta_x \\
X_{P_2 I_0} &= \alpha_x X_{P_2 I_1} + \beta_x X_{P_2 I_2} + \gamma_x X_{P_2 I_3} + \delta_x \\
X_{P_3 I_0} &= \alpha_x X_{P_3 I_1} + \beta_x X_{P_3 I_2} + \gamma_x X_{P_3 I_3} + \delta_x \\
X_{P_4 I_0} &= \alpha_x X_{P_4 I_1} + \beta_x X_{P_4 I_2} + \gamma_x X_{P_4 I_3} + \delta_x
\end{aligned}
$$

The procedures that solve these equations are as follows:

$$
\Delta = \begin{vmatrix}
X_{1_{p0}} & X_{2_{p0}} & X_{3_{p0}} & 1 \\
X_{1_{p1}} & X_{2_{p1}} & X_{3_{p1}} & 1 \\
X_{1_{p2}} & X_{2_{p2}} & X_{3_{p2}} & 1 \\
X_{1_{p3}} & X_{2_{p3}} & X_{3_{p3}} & 1
\end{vmatrix}
\qquad
\Delta_\alpha = \begin{vmatrix}
X_{0_{p0}} & X_{2_{p0}} & X_{3_{p0}} & 1 \\
X_{0_{p1}} & X_{2_{p1}} & X_{3_{p1}} & 1 \\
X_{0_{p2}} & X_{2_{p2}} & X_{3_{p2}} & 1 \\
X_{0_{p3}} & X_{2_{p3}} & X_{3_{p3}} & 1
\end{vmatrix}
$$

$$\Delta_\beta = \begin{vmatrix} X_{1_{p0}} X_{0_{p0}} X_{3_{p0}} 1 \\ X_{1_{p1}} X_{0_{p1}} X_{3_{p1}} 1 \\ X_{1_{p2}} X_{0_{p2}} X_{3_{p2}} 1 \\ X_{1_{p3}} X_{0_{p3}} X_{3_{p3}} 1 \end{vmatrix} \quad \Delta_\gamma = \begin{vmatrix} X_{1_{p0}} X_{2_{p0}} X_{0_{p0}} 1 \\ X_{1_{p1}} X_{2_{p1}} X_{0_{p1}} 1 \\ X_{1_{p2}} X_{2_{p2}} X_{0_{p2}} 1 \\ X_{1_{p3}} X_{2_{p3}} X_{0_{p3}} 1 \end{vmatrix} \quad \Delta_\delta = \begin{vmatrix} X_{1_{p0}} X_{2_{p0}} X_{3_{p0}} X_{0_{p0}} \\ X_{1_{p1}} X_{2_{p1}} X_{3_{p1}} X_{0_{p1}} \\ X_{1_{p2}} X_{2_{p2}} X_{3_{p2}} X_{0_{p2}} \\ X_{1_{p3}} X_{2_{p3}} X_{3_{p3}} X_{0_{p3}} \end{vmatrix}$$

$\alpha = \Delta_\alpha/\Delta$, $\beta = \Delta_\beta/\Delta$, $\gamma = \Delta_\gamma/\delta = \Delta_\delta/\Delta$

After solving these equations,we can use α_x, β_x , γ_x and δ_x values to predict the x coordinate value of any point in the unknown image from the corresponding points in the three model images. Then these predicted values can be used to compare to the original x coordinate values in unknown image. If the difference between them is less than a certain threshold, these two points match with each other and if all feature points match, the conclusion can be made that the unknown image matches the image models.

The above discussion is only concerned about x coordinates, we can build similar equations for y coordinate, by using same method as that used for x coordinate, producing another set of constrants: α_y, β_y, γ_y and δ_y to predict the y coordinate value of any point in the unknown image.

3 Some Issues

From the description of LCM in last section, we can see during the match procedures, a criterion is needed to determine how big the difference between the predicted point and the original one is acceptable, that is, how big the threshold should be. The threshold selection is critical and it directly determines the match result. If the threshold is too big, some objects which actually do not match the model object will be considered to be matched; and if the threshold is too small, some objects which should be matched are unmatched. How to select an appropriate threshold is very difficult and still under research. Although some methods have come out, they are either too complex, having to do large amount of calculation, or only applying to certain special case. Up to date, the most popular and simplest method to be used is still an experimental approach, which is used in my design. in this method, a sequence of thresholds are supplied, user can choose different thresholds for various objects, which gives more flexibility to users.

Another important issue in Linear Combination Method is the feature points. How to find out the corresponding feature points and how many feature points being necessary are crucial in Linear Combination Method. There are some methods already existing, such as the approach that keeps track of the intermediate snapshots between each pair of images that is to be in the model or the method that sets up correspondence between sets of points instead of single point.

In our system, the hough transform was used and the steps to find feature points are as follows:

- Performing edge detection on original image

- Using a distributed hough transform to find the lines in the edge image
- Building a set of pairs of lines and finding all intersections, and if the intersection is on the edge, labeling it as a feature point.

Now the question is that how many feature points are at least necessary. For model images, as it is mentioned before, four feature points are enough; however, for the unknown image, four points are used to learn information from unknown object and to calculate α, β, γ and δ constraints, therefore, we can not use α, β, γ and δ, which come from these four points, to predict the coordinate values of these four points because the result will always match and this does not make any sense. Hence, at lease one more feature point is needed so that we can use α, β, γ and δ values to predict the coordinate value of this point.

4 LCM on Articulated Objects

Articulated object consists of several parts, each of which can rotate independently and has its particular α, β, γ and δ value. Therefore, linear combination computation has to be done on each of them, that is, every part is considered to be an independent object and is recognized separately. After recognition of each part, all the results are combined together. If every part iof the object is matched, the unknown object matches with the model images, otherwise, they do not match.

5 Experimental Results

Based on the concepts in previous sections, some of the examples and experimental results are given in this section. As it is mentioned before, the Linear Combination Method applies to both rigid objects, which include both convex and cancave objects, and articulated objects with visible and invisible hinges. This will cover a large amount of objects in real world and that is why this method is so useful.
The following gives four kinds of objects and shows how the LCM works on them according to the experimental data.

A) Rigid Convex Object

Figure 4.1 shows the various views of two kinds of pyramid. (i),(ii) and (iii) are model images, which are the different views of a same pyramid. After hough transform, four feature points are selected to set up model information, which will be used as a template to match with all unknown input images. (iv) and (v) are two input unknown images. After hough transform, at lease five feature points, as labeled in Fig. 4.1, have to be selected, four of them are used to calculate the α, β, γ and δ values to learn from unknown images, and the last one of them is used to determine if the unknown images match with the model images.
The image which is shown in Fig. 4.1 (iv) is the same object as the models. After the LCM is applied to it, the result should be matched and the following data verify this result.

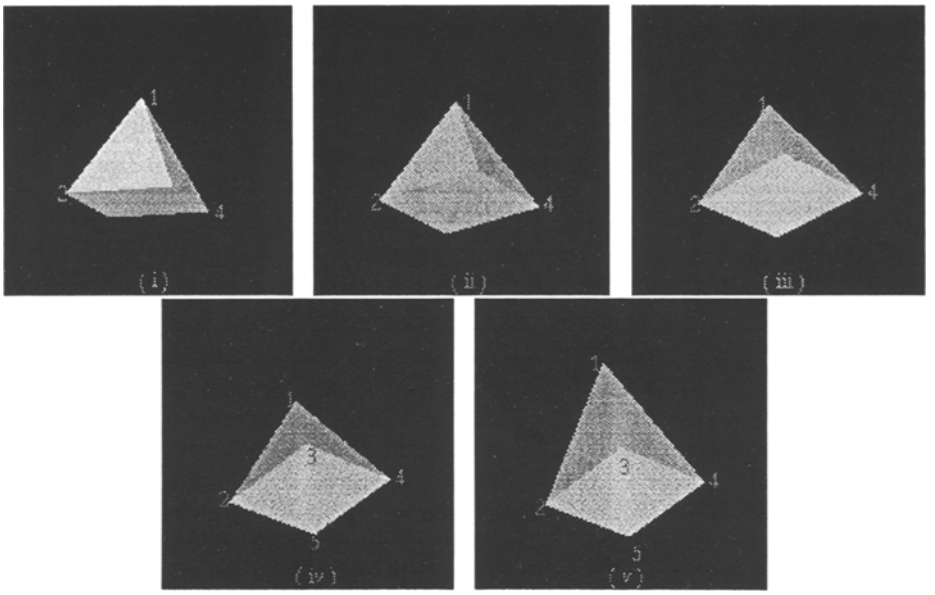

Figure 4.1 Rigid convex object recognition
(i), (ii) and (iii) are model images, (iv) is a different view
of model object, and (v) is a view of another different object

The feature points used on three model images are as follows:

Model1: (0.10,1.40), (-2.10,-1.20), (0.80,-1.00), (1.90,-1.70)
Model2: (-0.10,1.30), (-2.10,-1.30), (0.50,-0.50), (2.20,-1.50)
Model3: (-0.10,1.20), (-2.00,-1.40), (0.20,-0.10), (2.30,-1.10)

and the selected feature points of this unknown image are as follows:

(-0.20,1.2), (-2.0,-1.5), (0.0,0.0), (2.3,-0.9), (0.3,-2.4)

The match procedures are as follows:

α_x =-0.23, β_x=-0.01, γ_x=-0.06
α_y= 0.46, β_y=-1.32, γ_y=-0.01

The predicted x and y coordinate values are 0.2530 and -2.5315 and the absolute differences between the predicted value and the original one are 0.047 and 0.1315. If we select the threshold to be 0.2, both the differences are less than threshold, therefore, this unknown image matches with the model images. Compared to the image in (iv), the image which is shown in Fig. 4.1 (v) is the

pyramid with different height than the model and the following data demonstrate that it does not match the model images.
The feature points selected from this image are as follows:

(-0.3,2.3), (-2.0,-1.5), (0.0,0.0), (2.3,-0.9), (0.3,-2.4)

and

α_x = -0.68, β_x = 0.96, γ_x = 0.68, δ_x = -0.07
α_y = 1.87, β_y =-3.32, γ_y = 2.91, δ_y = 0.50.

The predicted values of x and y coordinates are 0.2592 and -2.7318. The absolute differences between the original one and the predicted one are 0.0408 and 0.3818. If we still select the threshold to be 0.2, it is obvious that the differences are greater than the threshold. Therefore, this input image does not match the model images.

B) Rigid Concave Object

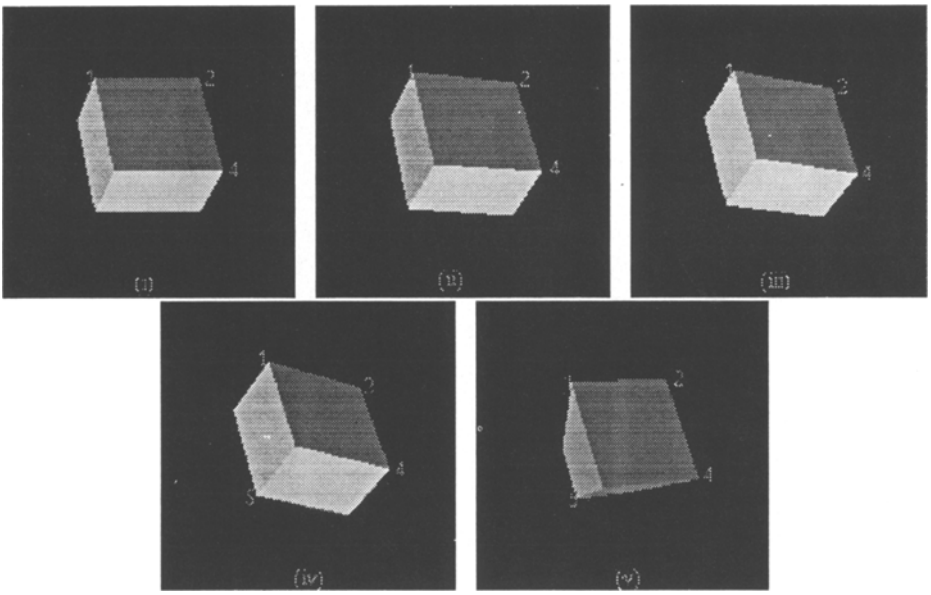

Figure 4.2 Rigid concave object recognition
(i), (ii) and (iii) are the images of model object
which is a box without cover. (iv) is the different
view of the same box, and (v) is different shape box

The recognition procedures for rigid concave objects are almost the same as those for convex objects, except that at lease one of the feature points for model images and unknown images has to be visible inside point, which is labeled 3 in above picture.

Fig. 4.2. shows three models and two unknown images.

C) Articulated Objects

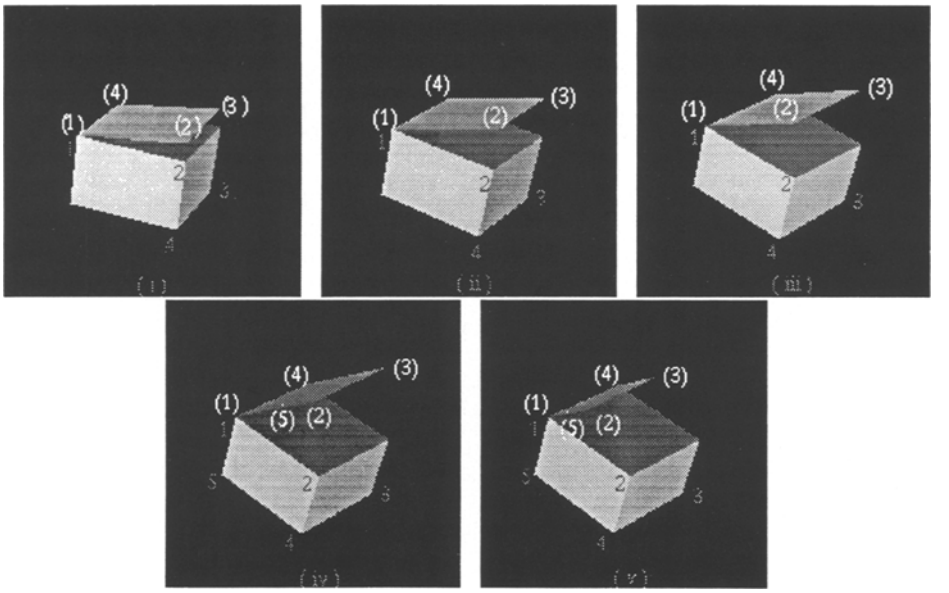

Figure 4.3 Articulated object recognition
(i), (ii), and (iii) are model images of
the closet, (iv) is another view of the same
closet, and (v) is the image of a different closet

The objects in Fig. 4.3 are kinds of closets and their doors can be rotated along their hinges. The three model images in (i), (ii) and (iii) are set up by rotating both main part and articulated part separately. Fig. 4.3 (iv) shows the different view of the same object and the following data verify that it matches the model images.
The match procedures on the main part are described as follows:
The feature points on the main parts of the model images are as belows:

Model1: (-1.90,0.40), (1.00,-0.20), (1.70,-1.10), (0.70,-2.10)
Model2: (-2.00,0.60), (0.70,-0.50), (1.70,-1.20), (0.30,-2.20)
Model3: (-2.00,0.70), (0.40,-0.70), (1.70,-1.30), (0.00,-2.40)

and the selected feature points on the main part of this unknown image are

(-2.0,0.8), (0.3,-0.8), (1.7,-1.3), (-0.2,-2.4), and (-2.3, -0.8).

After the equations are solved, the constrain values are as follows:

$\alpha_x = -1.12, \beta_x = 1.78, \gamma_x = 0.31, \delta_x = 0.05$
$\alpha_y = -0.32, \beta_y = 0.09, \gamma_y = 1.21, \delta_y = 0.03$

the predicted x and y coordinate values are -2.4030 and -0.7106 and the absolute differences between the predicted value and the original one are 0.1030 and 0.0894. If we select threshold to be 0.6, both the differences are less than threshold, so far the main part of this unknown image matches those of the model images.
The match procedures on the articulated portion are described as follows:
The feature points on the articulated portion of the model images are as follows:

Model1: (-1.90,0.40), (1.00,0.30), (2.00,1.20), (-0.80,1.30)
Model2: (-2.00,0.60), (0.70,0.60), (2.10,1.40), (-0.40,1.40)
Model3: (-2.00,0.70), (0.40,0.80), (2.20,1.70), (-0.10,1.60)

and the feature points on the articulated portion of this unknown image are

(-2.0,0.8), (0.1,1.2), (2.2,2.1), (0.0,1.7), and (-0.95,1.0).

After calculation, the constrain values are as follows:

$\alpha_x = 0.62, \beta_x = -2.00, \gamma_x = 2.38, \delta_x = -0.07,$
$\alpha_y = 0.33, \beta_y = 0.67, \gamma_y = 1.00, \delta_y = 1.60$

the predicted x and y coordinate values are -1.4835 and 1.2656 and the absolute differences between the predicted value and the original one are 0.5335 and 0.2656. If we select threshold to be 0.6, both the differences are less than threshold, so far the articulated portion of this unknown image also matches those of the model images. Therefore, the conclusion can be made that this unknown image matches the model images.
Fig. 4.3 (v) is a closet which has the same main part as that of model object, however, its door is different from that of the model one: it has a various size. Therefore, when this input image is recognized, the main part should match the models and the articulated portion should not. Hence, the entire image will not match model images. The followings are the recognization data which demonstrate the above statement.
Because the main part of this unknown image is the same as that in (iv) and from the above calculation, we know that it matches the main parts of the model images.
Next, let us focus on the articulated portion recognition.
The feature points on the articulated portion of this unknown image are

(-2.0,0.8), (-1.0,1.0), (1.1,1.9), (0.0,1.7), and (-1.5,0.9).

After calculation, the constraints are as follows:

$\alpha_x = 0.26, \beta_x = -1.68, \gamma_x = 2.62, \delta_x = -0.61,$
$\alpha_y = 0.50, \beta_y = -0.92, \gamma_y = 0.67, \delta_y = 0.40$

the predicted x value is -0.0273 and the absolute difference between the predicted value and the original one is 1.4727. If we select threshold to be 0.6, both the difference is much greater than threshold, so far the articulated portion of this unknown image does not match that of the model images.
Therefore, the conclusion is that this unknown image does not match the model images, as stated before.

6 Discussion and Conclusion

The Linear Combination Method presented here is based on the concept of the original LCM. In this paper, some important issues are discussed about this method, especially on how to use it on an articulated object. Also, some learning samples are given to show the whole recognition procedures. The Linear Combination Method does not need any explicit three dimensional description and is very simple and easy to implement. Although there are still some problems which have not had good solutions, such as how to get feature points effectively and the threshold selection method , it still is a good and useful method that may become one of the major methods in object recognition field because of its simple computation and few learning samples.

Acknowledgment

The authors woud like to thank the College of Computer Sciences for providing an excellent laboratory and research environment, without which this article is impossible.

References

1. L. Baird and P. Wang, „3D Object Perception Using Gradient Descent", Int. Journal of Mathematical Imaging and Vision (IJMIV), 5, 111-117, 1995
2. R. Basri, „ Viewer-Centered Representations in Object Recognition: A Computational Approach", Handbook of Pattern Recognition and Computer Vision, pp. 863-882, WSP (1993)
3. R. Brooks, „Symbolic reasoning around 3-d models and 2-d images", Arti.Int., 17, 285-348, 1981
4. R. Goldberg and D. Lowe, „Verification of 3-d parametric models in 2-d image data", Proc. IEEE Workshop on Computer Vision, 255-267, 1987
5. W.E.L. Grimson and T. Lozano-Perez, „Localizing overlapping parts by searching the interpretation tree", IEEE-PAMI, 9(4), 469-482, 1987
6. T. Marill, „Emulating the human interp. of line-drawings as 3d objects", IJCV, v6-2, 1991, 147-161
7. P. Wang, „Recognizing 3D Articulated Line-Drawing Objects", 120/SPIE vol. 2056 Intelligent Robots and Computer Vision X11 (1993) pp120-131

8. P. Wang, „3D Line Image Analysis - A Heuristic Parallel Approach with Learning and Recognition", Info. Sciences v81, 1994 pp. 155-176
9. P. Wang, „A Heuristic Approach for 3D Articulated Line-Drawing Object Pattern Representation and Recognition" , Advances in Imaging Technologies (ed by E.Dougherty), Marcel Dekker, 1994, 197-221
10. P. Winston, Artificial Intelligence, Cha. 26 „Recognizing Objects", pp. 531-551, Addision Wesley, 1994