A Corner Matching Algorithm with Uncertainty Handling Capability

Kil-jae Lee* and Zeungnam Bien**

* Production Engineering Research Center, LG Electronics Inc.
** Department of Electrical Engineering
Korea Advanced Institute of Science and Technology
373-1, Kusong-dong, Yusong-Ku, Taejeon, 305-701, KOREA

***Abstract*— An efficient corner matching algorithm is developed to minimize the amount of calculation. To reduce the amount of calculation, all available information from a corner detector is used to make model. This information has uncertainties due to discretization noise and geometric distortion, and this is represented by fuzzy rule base which can represent and handle the uncertainties. From fuzzy inference procedure, a matched segment list is extracted, and resulted segment list is used to calculate the transformation between object of model and scene. To reduce the false hypotheses, a vote and re-vote method is developed. Also an auto tuning scheme of the fuzzy rule base is developed to find out the uncertainties of features from recognized results automatically. To show the effectiveness of the developed algorithm, experiments are conducted for images of real electronic components.**

Key words: fuzzy logic, pattern classification, corner point matching, hypotheses and verify.

I. Introduction

To recognize and locate objects, a matching procedure should be followed after a local feature extraction procedure. In this paper, corners are selected as the local features. These corners have the information enough to recognize and locate objects. The model object also has corner points, and by matching these corner points with corner points in a real image, it is possible to recognize and locate objects.

Fig.1 shows an example of the corner matching problem. Fig.1(a) shows corners of model object, and Fig.1(b) shows corners of a scene. Fig.1(c) shows corners of the model object which are hide in the scene. As shown in Fig.1, matching a template of points in a model with the points in a real image is difficult to solve since there are many exceptional cases which should be considered[1]. First, great many feature points may be present because of multiple instances of the chosen type of the object in the image. Second, additional points may be present because of noise or clutter from irrelevant objects and structure in the background. Third, certain points that should be present are missing because of noise or occlusion, or because of defects in the object being sought. And finally, feature values from corner detector have the uncertainties associated with their numerical properties. In designing the matching procedure, the above points should be taken into account.

Many matching algorithms have been developed to reduce the amount of calculations because the amount of calculations is increasing explosively with an increase of local features. Maximal clique finding algorithm[1] is a mathematical method to solve the corner matching problem. The

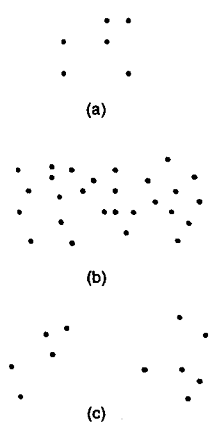

Fig. 1. Corner matching problem: (a) corners of model object, (b) corners of scene, (c) recognized result

corner matching problem can be formulated as the problem of subgraph-subgraph isomorphism. An association graph is introduced to solve this problem in systemically. Association graph is made of the nodes which represent feature assignments, and arcs joining nodes which represent pairwise compatibilities between assignments. By finding maximal clique of the association graph, matching problem can be solved. But the problem of finding maximal clique is not simple because this problem is NP-complete problem.

The graph matching process can be formulated as an optimization problem. An energy function is derived so that this function represents the constraints that the nodes in the two graphs have to satisfy in order to find the best match. A 2-D binary Hopfield neural network[2] or relaxation algorithm[3] can be used to minimize the energy function. But because these methods need iterative operations, there must be considerations about convergence, and it may take long time.

Another one is hypotheses and verify method[4-6]. By using privileged features and similarity measure, a hypotheses can be generated. By calculating match confidence, the hypotheses can be verified. The hypotheses and verification method can reduce the amount of calculations

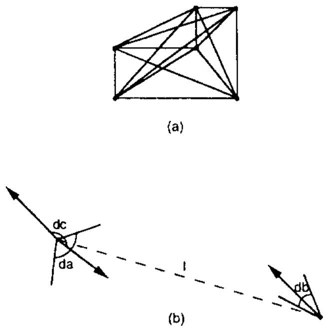

Fig. 2. The features of segment: (a) all possible segments of the model, (b) the features of segment

remarkably. But the hypotheses generation scheme is complex, and false hypotheses may cause the system to spend a long time to verify the hypotheses. Therefore, the design of the hypotheses generation scheme is important.

In this paper, an effective algorithm to solve the matching problem will be presented. The paper is organized as follows. The proposed corner matching algorithm is described in Section 2. Experiments to show the effectiveness of the proposed algorithm are conducted in Section 3. The summary and conclusion follows in Section 4.

II. Algorithm for Corner Matching

A. Hypotheses Generation Procedure

A segment is consisted of two corners. Let's assume that the corner points of the model object are given by a set $\{MC_1, MC_2, \ldots, MC_m\}$. The information of a corner MC_i is given by a vector $(x_i, y_i, \alpha_i, d\alpha_i)$, where x_i and y_i are position of corner, α_i is angle of corner, $d\alpha_i$ is angle change of corner. From two corners, MC_i and MC_j, a feature vectors of a segment MS_k connecting these corners can be calculated. The feature vector of a segment MS_k is represented as (l_k, da_k, db_k, dc_k), where l_k is length of the segment, da_k and db_k are two angle changes of two corners and dc_k is angle difference between two corners as shown in Fig.2. The features of the segment can be calculated as follows:

$$l_k = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{1}$$

$$da_k = d\alpha_i \tag{2}$$

$$db_k = d\alpha_j \tag{3}$$

$$dc_k = |\alpha_i - \alpha_j| \tag{4}$$

By connecting the corner points of the model object, a model segments set $\{MS_1, MS_2, \ldots, MS_P\}$ is acquired, where $P = \frac{m\cdot(m-1)}{2}$. To represent the model object, the feature vectors of the model segments are used. But these feature vectors have uncertainties due to discretization noise, geometric distortion and feature calculation error. To accommodate these uncertainties, the fuzzy logic is used to represent these feature vectors. The feature vector of a segment MS_i can be represented as (L_i, DA_i, DB_i, DC_i), where L_i, DA_i, DB_i and DC_i are fuzzy numbers, and these fuzzy numbers are defined by the triangular membership functions $\mu_{L_i}(l)$, $\mu_{DA_i}(da)$, $\mu_{DB_i}(db)$ and $\mu_{DC_i}(dc)$. By using these feature vectors, the model object can be represented as P fuzzy rules as follows.

If length is L_i and (two angle changes are DA_i or DB_i) and angle difference is DC_i, then the segment is MS_i. $i=1,2,\ldots,P$.

So, the model object can be represented by assigning the fuzzy rule into each segment.

By using the model which is represented by the fuzzy rules, a matched segment list can be extracted. A matched segment pair is a pair of a segment of the model and a segment of the scene, where the segment of the scene is regarded as the segment of the model. After corner detection, corners of the scene are acquired as a set $\{SC_1, SC_2, \ldots, SC_n\}$. From this scene corner set, scene segments list $\{SS_1, SS_2, \ldots, SS_Q\}$ can be acquired, where $Q = \frac{n\cdot(n-1)}{2}$. For a scene segment SS_j, a segment feature vector (l_j,da_j,db_j,dc_j) can be calculated with the same calculating procedure of the model segment feature vector. After the fuzzy inference procedure is done for the segment feature vector, a matching degree between the model segment and the scene segment is calculated. To calculate the matching degree between the model segment MS_i and the scene segment SS_j, the following equation is used.

$$\mu_{MS_i}(SS_j) = \mu_{L_i}(l_j) \cdot max\{\mu_{DA_i}(da_j), \mu_{DB_i}(db_j)\} \cdot \mu_{DC_i}(dc_j). \tag{5}$$

Here, $\mu_{MS_i}(SS_j)$ represents the matching degree between the model segment MS_i and the scene segment SS_j. This fuzzy inferencing procedure is summarized in Fig.3.

After calculating all matching degrees of the model segments and the scene segments, a matched segment list(MSL) is acquired by using thresholding.

$$MSL = \{(SS_j, MS_i) | 1 \le j \le Q, 1 \le i \le P, \mu_{MS_i}(SS_j) \ge T\}. \tag{6}$$

B. Transformation and Clustering

There may be multiple objects in a scene. Each object is consisted of the segments having same transformation parameters between scene segments and matched model segments. So, to detect multiple objects in a scene, most algorithms use a clustering method called as pose clustering or hypotheses accumulation or generalized hough transform after calculation of transformation parameters[11]. This recognition method is consisted with two stages: calculation stage of pose and clustering stage of pose. By processing the above stages, the objects in a scene can be detected in separate.

Pose of the objects in a scene can be represented as the transformation parameters. The coordinate transforma-

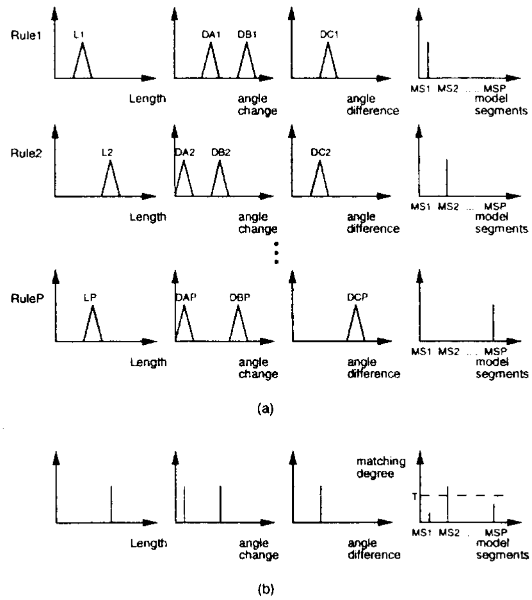

Fig. 3. Fuzzy inference procedure to extract matched segment list

tion for a pair of matched segment can be represented by a tuple (θ, tx, ty) in a 3-D space, where θ is the angle of rotation, and tx and ty are the translations in X axis and Y axis respectively. Let two corners in the model segment be denoted by $MC_1(x_{m1},y_{m1})$ and $MC_2(x_{m2},y_{m2})$, and the two corners in the scene segment by $SC_1(x_{s1},y_{s1})$ and $SC_2(x_{s2},y_{s2})$. Since the correspondence between the model corners and the scene corners is not known, there can be two possible transformations (θ_1,tx_1,ty_1) and (θ_2,tx_2,ty_2) between the pair of matched segment. The corner points MC_1 and MC_2 can be mapped to the scene corner points SC_1 and SC_2 by the following transformations:

$$x_{s1} = tx_1 + \cos\theta_1 \cdot x_{m1} - \sin\theta_1 \cdot y_{m1} \tag{7}$$

$$y_{s1} = ty_1 + \sin\theta_1 \cdot x_{m1} + \cos\theta_1 \cdot y_{m1} \tag{8}$$

$$x_{s1} = tx_2 + \cos\theta_2 \cdot x_{m2} - \sin\theta_2 \cdot y_{m2} \tag{9}$$

$$y_{s1} = ty_2 + \sin\theta_2 \cdot x_{m2} + \cos\theta_2 \cdot y_{m2} \tag{10}$$

$$\theta_2 = \theta_1 + \pi \tag{11}$$

The same mapping can be obtained for the point MC_2 using the same formulas given previously. From the above equations, two transformation parameter set, (θ_1,tx_1,ty_1) and (θ_2,tx_2,ty_2), can be calculated.

For each pair of matched segments, its coordinate transformations can be tabulated in a 3-D accumulator. This accumulator behaves like the hough transform, the correct matched pairs will form large clusters in the transformation space (θ, tx, ty) and the false matched pairs will be scattered in several small clusters.

The accumulator is examined to find the locations of the large values to be used as the centers of the clusters. To find the large clusters, a simple clustering method is used. The clustering method applied here is the one-pass clustering algorithm[8]. If the distance between the locations of the centers of the clusters and a point in the transformation space is less than some threshold, then the point is considered to belong to the clusters. The neighborhood sizes for the angle and the translation coordinates (θ, tx, ty) are different. The selected cluster neighborhood sizes for the angle and the translations were based on the experimental experience. The clusters having elements greater than some threshold can be regarded as the clusters having the model object.

C. Reduction of False Hypotheses

Although the cluster has the elements greater than some threshold, the cluster may be composed of wrong matched segments. Such cluster results false hypotheses, and this may cause the long processing time. To reduce the false hypotheses, a concept of inter-segment is introduced. An inter-segment is consisted of two segments and defined by a segment which connects from the center of a segment to the center of the other segment. The feature vector of the inter-segment IS is represented as (ID, IA, IB), where ID is the length of the inter-segment, IA and IB are two angles between the inter-segment and two segments as shown in Fig.4.

The inter-segment can be used to avoid the noise problem of the generalized Hough transform. The generalized Hough transform can not distinguish transformation parameters resulting from wrong features near to the correct features. To distinguish such transformation parameters resulting from wrong features, a purifying procedure is necessary.

Let two matched segment pairs in a cluster be given by (MS_i,SS_p) and (MS_j,SS_q) and the inter-segment between the model segments by IS^M_{ij} and the inter-segment between the scene segments by IS^S_{pq}. When these segment pairs come from the correct features, the feature vectors of the inter-segments, IS^M_{ij} and IS^S_{pq}, should be similar. To use this fact, a voting procedure is introduced. If the feature vectors of IS^M_{ij} and IS^S_{pq} are similar, scene segments SS_p and SS_q vote to each other. The above procedure is applied for all matched segment pairs in the cluster. If there are N scene segments which come from the correct features in the cluster, the number of votes of the scene segment from correct features will be greater than N. And the number of votes of the scene segment from wrong features may be less than N.

But in some cases, the number of votes of the scene segment from wrong features can be greater than N due to the votes from the other segments which come from wrong features. So, vote and re-vote procedure is introduced to avoid such situation. Since it is rare case that the objects in a scene are exactly overlapped, a cluster can be assumed to contain only one object. This means that the number of scene segments matching with a model segment should

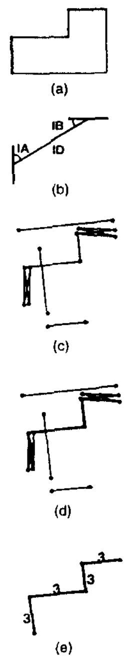

Fig. 4. Vote and re-vote method: (a) model object, (b) feature vector of inter-segment, (c) matched segments in a cluster, (d) result from vote and purify procedure, (e) result from revote procedure

be less than 2. By using this fact, a purifying procedure to select one scene segment which matchs best with a model segment is done. From the above voting procedure, the scene segment having the largest number of votes for each model segment is taken as the matching segment with the model segment. The other matched segment pairs which have not the largest number of votes are regarded as the pairs from wrong features and rejected from a list of the cluster.

After the purifying procedure is done, the same voting procedure described above is done again to the purified clusters. Because of the above purifying procedure, if there are N segments which come from correct features in the cluster, the number of votes of a scene segment from wrong features must be less than N. So, by thresholding with N, the noise problem of the generalized hough transform can be solved.

D. Verification Procedure

A verification procedure is necessary to test whether the generated hypotheses is correct or not. A hypotheses consists of an model object and a pose parameters in the scene. By using these transformation parameters, the model object can be transformed in the scene. By calculating correlation between the scene object and the transformed model object, it is possible to verify the hypotheses.

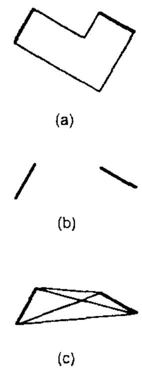

Fig. 5. Reconstruction of features of possible model segments: (a) object in scene, (b) two matched segments, (c) six reconstructed segments

E. Auto Tuning Algorithm of Input Membership Functions

The proposed system can handle the uncertainties of local features by fuzzy input membership functions. To design such a system, it is necessary to know the uncertainties of local features. M. D. Wheeler and K. Ikeuchi[9] use a large set of ray-traced images of the object to find out the uncertainties of features. They modeled the sensing process and simulated with the model in various view points. But this method may be time-consuming and very expensive process.

A feedback mechanism to find out the uncertainties of features is developed. From the recognized results, values of input features can be calculated as shown in Fig.6. At first, reconstruction of possible model segments is done. As shown in Fig.5, six reconstructed segments can be extracted from two matched segments of the recognized results. By using the feature vectors of these reconstructed segments, input membership functions can be adjusted. After the above procedure is done for all of the recognized results, tuned input membership functions can be acquired.

Let current number of feature values be n, and mean value of feature values be denoted by M_n, and standard deviation of feature values by σ_n. When a new feature value f_{n+1} is added, new mean value M_{n+1} and new standard deviation σ_{n+1} can be calculated by following equations.

$$\sigma_{n+1}^2 = \frac{n \cdot \sigma_n^2}{n+1} + \frac{(M_n - f_{n+1})^2 \cdot (2 \cdot n^2 + n)}{(n+1)^3} \qquad (12)$$

$$M_{n+1} = \frac{n \cdot M_n + f_{n+1}}{n+1} \qquad (13)$$

III. Experimental Results

To test the effectiveness of the developed corner matching algorithm, an experiment is conducted. Test images are consisted of 50 images which contain 183 electronic components. Fig.7 shows the result of the experiment. Fig.7(a)

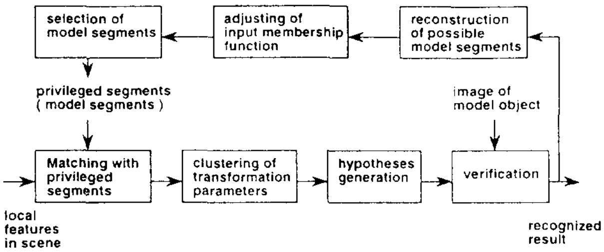

Fig. 6. Block diagram of auto tuning algorithm

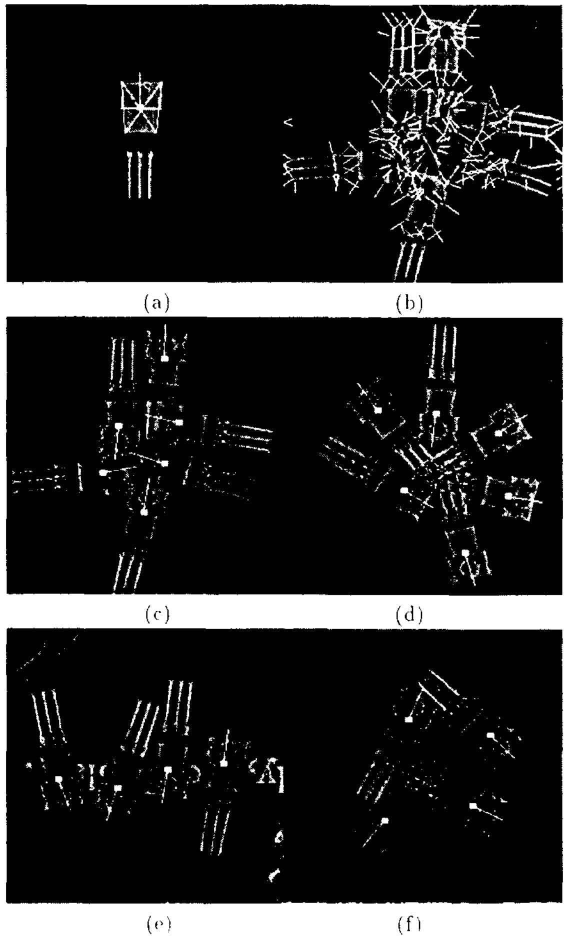

Fig. 7. Recognition results using developed algorithm. (a) model segments, (b) detected corners, (c),(d), (e) and (f) recognition results

shows 8 model segments. Fig.7(b) shows information of extracted corners by using the gray level corner detector and a localization procedure. As shown in Fig.7(b), corners of scene objects have a lot of uncertainties due to discretization noise and shadow. The recognized results are represented by arrows on scene image. Fig.7(c) and (d) show that the developed corner matching algorithm works well in the presence of partial occlusion and shadow. Also Fig.7(e) and (f) show that the algorithm works well even in the presence of non-uniform background.

To test the effectiveness of the auto tuning algorithm, an experiment has been conducted. Fig.8 shows the results with iterations of auto tuning method. An iteration is consisted of processings of 50 images. As shown in Fig.8(a), recognition rate is greater than 95 percent. Number of hypotheses and matched segments are reduced into about 3 times as the auto tuning method is progressing. So, the processing time is much faster than no auto tuning method is applied. The matching time of 200 corners is reduced into about 2.5 seconds by using INDIGO 2 workstation. If a dedicated DSP is used, the processing time will be reduced further.

IV. Summary and Conclusion

An effective corner matching algorithm is developed. To reduce the matched segment list effectively, it is shown that all the available information of segment can be used to represent the feature vector of segment. Using the resulting matched segment list, the position and orientation of each object can be calculated effectively. By adjusting the input membership functions of the fuzzy rule base, the amount of uncertainties in feature values can be incorporated into the developed vision system. Since it is difficult to know the amounts of uncertainties, an automatic tuning algorithm of the fuzzy rule base is developed. This corner matching procedure can be easily extended to the matching of other local features such as holes and curves.

References

[1] E. R. Davies, *Machine Vision : Theory, Algorithms, Practicalities*, Academic Press, 1990.

[2] N. M. Nasrabadi and W. Li, "Object recognition by a Hopfield neural network", *IEEE Trans. System, Man and Cybern.*, vol. 21, no. 6, p1523-1535, 1991.

[3] W. S. Rutkowski, "Recognition of occluded shapes using relaxation", *Comput. Vision Graphics Image Process.*, vol. 19, p111-128, 1982.

[4] N. Ayache and O. D. Faugeras, "HYPER: A new approach for the recognition and positioning of two-dimensional objects", *IEEE Trans. Patt. Anal. Machine Intell.*, vol. 8, no. 1, p44-54, 1986.

[5] M. W. Koch *et al.*, "Using polygons to recognize and locate partially occluded objects", *IEEE Trans. Patt. Anal. Machine Intell.*, vol. 9, no. 4, p483-494, 1987.

[6] E. H. Bouyakhf, "An approach for recognizing and locating overlapping industrial parts", *Intern. Journal of Pattern Recog. and Artificial Intell.*, vol. 2, no. 1, p673-689, 1988.

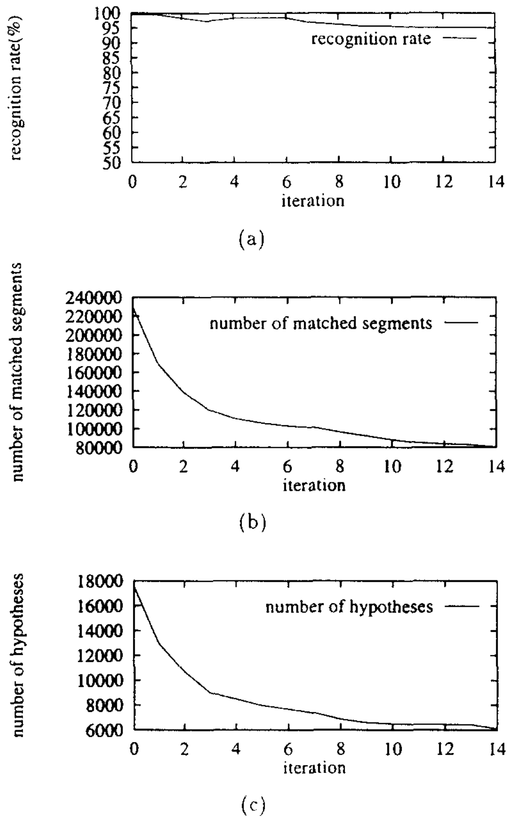

Fig. 8. Results of auto tuning algorithm (a) recognition rate (b) number of matched segment (c) number of hypotheses

[7] W. Pedrycz, *Fuzzy Control and Fuzzy Systems*, John Wiley and Sons Inc., 1989.

[8] J. A. Richards, *Remote Sensing Digital Image Analysis*, Springer-Verlag, 1994.

[9] M. D. Wheeler and K. Ikeuchi, "Sensor Modeling, Probabilistic Hypothesis Generation, and Robust Localization for Object Recognition", *IEEE Trans. Patt. Anal. Machine Intell.*, vol. 17, no. 3, p252-265, 1995.

[10] W. Pedrycz, "Fuzzy sets in pattern recognition: methodology and methods", *Pattern Recognit.*, vol. 23, no. 1, p121-146, 1990.

[11] G. Stockman, "Object Recognition and Localization via Pose Clustering", *Comput. Vision Graphics Image Process.*, vol. 40, p361-387, 1987.