A modified genetic algorithm for image segmentation based on feature clustering

Miki AOYAGI† and Kumiko Tsuji‡

† Department of Mathematics
Sophia University,
7-1 Kioi-cho, Chiyoda-ku, Tokyo,
102–8554, JAPAN
E-mail: miki-a@sophia.ac.jp

‡ Fukuoka Junior College
Teikyo University,
4-3-124 Shin-Katsutachimachi, Ohmuta
836, JAPAN E-mail: c74206g@wisdom.cc.kyushu-u.ac.jp

Abstract

This paper proposes a new genetic algorithm (GA) based on feature clustering with an energy function for obtaining optimal segmentation. In the proposed algorithm, which we call MGA, the length of each genome is the number of features and each individual (genome) represents one assignment of the input-features to output layers. This algorithm first performs a sequence of operations for creating initial individuals (genomes) in the first generation. We propose four different kinds of special mutation operations. It is shown that MGA performs very well and is an improvement on the standard GA method for solving the contour grouping problem. It is also shown that MGA works more reliably than the competitive-layer model(CLM) [1] which is one of methods based on an energy function. MGA removes the difficulties of annealing problem and parameter adjustments which are necessary in CLM.

1. INTRODUCTION

Image segmentation based on feature clustering is performed by labeling features in an input image with a small number of labels such that features belonging to the same cluster have the same label and features belonging to different clusters have different labels.

Spin-lattice models are one of approaches to the image segmentation[2]. Every neuron of the neural net is mapped onto one spin element in the lattice and each feature is represented by a spin variable that attains one of a discrete set of spin states. These models use energy or cost functions that characterize stable output states as their minimums.

Similar to the spin-lattice model, the competitive-layer model (CLM) was introduced in [1] for the image segmentation based on an energy function. In CLM, each output layer is attributed to each cluster from the input layer, i.e., the image data of the input layer are segregated and such segmented data appear separately in the output layers. The dynamics move by not only interactions within each output layer but also winner-take-all (WTA) interactions among the output layers.

GA has been applied successfully to many problems of searching for an approximate global minimum of objective functions. Therefore, it is natural to consider how to apply GA to the feature clustering problem. Although there are some applications of GA, including genetic programming to the image segmentation, only a small number of applications of GA to the feature clustering problem have been reported [3]. The method in [3] is concerned with the problem of partitioning a feature group into homogeneous ones based on some measure but which are not suitable for application to contour grouping.

This paper proposes a new genetic algorithm with an energy function for solving the feature clustering problem. We call the proposed algorithm the modified genetic algorithm (MGA). Among GA operations, we propose four different kinds of special mutations for the problem. We will show later that our model works properly and it is difficult to get the ideal segmentation using the standard method of GA which is the simplified MGA without the creation of initial individuals and special mutations.

2. MGA ALGORITHM

In this section, we introduce our MGA algorithm.

Energy function (Fitness function)

Consider the problem of clustering R features which are denoted by parameter vector g_r, $1 \leq r \leq R$.

Let L be the number of labels and consider L output layers with R (the number of features) variables in each layer. It is unnecessary to know the number of clusters of features in the input image since the number of the output layers can be allowed to be more than the number of clusters. Denote rth variable in lth output-layer by v_{rl} which takes a value of 0 or 1. The feature

g_r having lth lavel is expressed by the activities $v_{rl} = 1$ and $v_{rl'} = 0$ for $l' \neq l$. Therefore, the feature clustering is achieved by determining v_{rl} so that each feature g_r is properly assigned to one of the output layers.

We denote a local interaction between features g_r and $g_{r'}$ by a scalar value $f_{g_r g_{r'}}$, which is a symmetric function of the feature parameters; $f_{g_r g_{r'}} = f_{g_{r'} g_r}$. A positive value $f_{g_r g_{r'}} > 0$ means that g_r and $g_{r'}$ are very likely to belong to the same cluster, and a negative value $f_{g_r g_{r'}} < 0$ means that g_r and $g_{r'}$ are likely to belong to different clusters. Therefore, we need to construct the fitness function such that an individual (genome) has a higher fitness value if g_r and $g_{r'}$ with $f_{g_r g_{r'}} > 0$ have the same label, or g_r and $g_{r'}$ with $f_{g_r g_{r'}} < 0$ have different labels. While the algorithm is running, g_r and $g_{r'}$ with $f_{g_r g_{r'}} > 0$ "cooperate" to stay on the same output layer and g_r and $g_{r'}$ with $f_{g_r g_{r'}} < 0$ "compete" to stay when g_r and $g_{r'}$ are on the same output layer. The absolute value of $f_{g_r g_{r'}}$ is the degree of competition or cooperation. Since the function $f_{g_r g_{r'}}$ determines a local interaction, its absolute value tends to 0 as the distance of the features g_r and $g_{r'}$ are longer.

To measure the fitness of assignment of the input features to the output layers, let us consider the following energy function, using the local interactions $f_{g_r g_{r'}}$ and global inhibition, which is similar to ones used in a Potts spin model [2] and in CLM [1].

Let

$$E = -\frac{1}{2}\sum_{l=1}^{L}\sum_{r=1}^{R}\sum_{r'=1,r'\neq r}^{R} f_{g_r g_{r'}} v_{rl} v_{r'l} \qquad (1)$$

$$+\frac{k}{2}\sum_{l=1}^{L}\sum_{r=1}^{R}\sum_{r'=1,r'\neq r}^{R} v_{rl} v_{r'l},$$

where $\sum_l v_{rl} = 1$, $v_{rl} \in \{0, 1\}$ and $k \geq 0$ is a control parameter that adjusts the strength of a global inhibition.

The first term means if features g_r and $g_{r'}$ are assigned to the same lth output-layer, i.e., $v_{rl} = v_{r'l} = 1$, then $-f_{g_r g_{r'}}$ is added, and if either g_r or $g_{r'}$ is not assigned to the lth output-layer, i.e., $v_{rl} = 0$ or $v_{r'l} = 0$, then $f_{g_r g_{r'}} v_{rl} v_{r'l} = 0$. That is, the energy function decreases when $f_{g_r g_{r'}} > 0$ and $v_{rl} = v_{r'l} = 1$, while it increases when $f_{g_r g_{r'}} < 0$ and $v_{rl} = v_{r'l} = 1$. The second term corresponds to the global inhibition which supports the separation among clusters.

The first step: Creation of the first individuals

We explain how to initially produce each individual with a genome for our GA.

Define an iteration for a step function v_{rl}

$$v_{rl}(\text{new}) = \begin{cases} 1 & \text{if } l = \min\{l_1 : w_{rl_1} \geq w_{rl_2} \forall l_2\} \\ 0 & \text{for the others,} \end{cases} \qquad (2)$$

where,

$$w_{rl} = \sum_{r'=1,r'\neq r}^{R} f_{g_r g_{r'}} v_{r'l}(\text{old}). \qquad (3)$$

Note that only the stock data of the local inhibitions are used. For initial values $v_{rl_r}(initial)$, one of output layers l_r, $1 \leq l_r \leq L$ is chosen randomly for each feature g_r, and we set $v_{rl}(initial)$ as

$$v_{rl_r}(initial) = 1, \quad v_{rl}(initial) = 0 \text{ if } l \neq l_r. \quad (4)$$

In each iteration, we modify only L variables v_{rl}, $l = 1, \cdots, L$, which correspond to one randomly selected feature g_r.

Let

$$E_{local} = -\frac{1}{2}\sum_{l=1}^{L}\sum_{r=1}^{R}\sum_{r'=1,r'\neq r}^{R} f_{g_r g_{r'}} v_{rl} v_{r'l}, \qquad (5)$$

which is the fist term of the energy function E (1).

Theorem By the iteration (2) and (3), we have $E_{local}(\text{new}) \leq E_{local}(\text{old})$, i.e., E_{local} makes a decreasing sequence.

By this method, only one variable v_{rl_r} is selected among corresponding variables v_{rl}, $l = 1, 2, \cdots, L$, and we get $v_{rl_r} = 1$ and $v_{rl} = 0$, $l \neq l_r$ in the direction of decreasing value of E_{local}. This iteration is continued until convergence, that is, until E_{local} reaches its local minimum. This method is used to determine each initial individual.

The second step : Applying GA

The purpose of the second step is to reach the globally minimal value of the energy function E (1) using GA and to get to the state of image processing where split contours generated by the first step are gathered and combined into proper groups. At the second step, we apply GA to get convergence at the minimal value of the energy function (1). In this step, the global inhibition is taken into account.

A genome of each individual contains a set of R parameters $(l_1, l_2, \cdots, l_R)$. Each parameter l_r represents the corresponding feature g_r. An individual represents one assignment of features to the output layers. The fitness function of GA is the energy function defined by (1), which is rewritten as follows.

$$E = -\frac{1}{2}\sum_{r=1}^{R}\sum_{r'=1,r'\neq r,l_{r'}=l_r}^{R} f_{g_r g_{r'}} + k\sum_{l=1}^{L}\sum_{r=1,l_r=l}^{R} v_{rl_r}.$$

We propose the following four kinds of special mutations.

- Mutation 1; r_1, r_2 $(1 \leq r_1 < r_2 \leq R)$ and l $(1 \leq l \leq L)$ are randomly selected and set as $l_{r_1}(i) = l_{r_1+1}(i) = \cdots = l_{r_2}(i) = l$.
- Mutation 2; r_1, r_2 $(1 \leq r_1 < r_2 \leq R)$ and l_1, l_2 $(1 \leq l_1, l_2 \leq L)$ are randomly selected and if $l_r(i) = l_1$, then set as $l_r(i) = l_2$ for $r_1 \leq r \leq r_2$.
- Mutation 3; l_1, l_2 $(1 \leq l_1, l_2 \leq L)$ are randomly selected and if $l_r(i) = l_1$, then set as $l_r(i) = l_2$ for all r.
- Mutation 4; l_1, l_2 $(1 \leq l_1, l_2 \leq L)$ are randomly selected. A number c is chosen such that $l_c = l_1$.

  Assign $l_c = l_2$.

  (u) For $1 \leq r \leq R$, if $l_r(i) = l_1$ and if there exists $l_{r'}(i) = l_2$ such as $f_{g_r g_{r'}} > 0$, then assign $l_r(i) = l_2$.

  The procedure (u) is continued until convergence.

Mutation 1 helps to combine nearby features into one group. Mutation 2 and Mutation 3 change the assignment of features from some layer to another layer and produce conditions to combine split contours. The difference between Mutation 2 and Mutation 3 is that Mutation 2 has a randomly selected range (r_1, r_2) for change. Mutation 4 causes separation of two contours distant from each other, since the algorithm changes only the assignment of all features having positive local interactions with each other from some layer to another layer.

Algorithm

The optimization procedure in GA consists of the following steps:

(0) Input M as the number of individuals, L as the number of output layers, R as the number of features, m_0 as the mutation probability, k as the global inhibition coefficient, and N as a stopping point number. Input a set of features g_r and interactions $f_{g_r g_{r'}}$.

(I) Create M initial individuals

gene$(i) = (l_1(i), l_2(i), \cdots, l_R(i))$.

For $1 \leq i \leq M$:

(1) Initialize integers $1 \leq l_j(i) \leq L, (j = 1, \cdots, R)$ with random values.

(2) For $1 \leq r \leq R$:

For $1 \leq l \leq L$: Put $w_{rl} = \sum_{l_{r'}(i)=l} f_{g_r g_{r'}}$

Update $l_r(i) = \min\{l : w_{rl} \geq w_{rl'} \text{ for all } l'\}$.

(3) The procedure (2) is continued until $l_r(i)$ do not change.

(II) Calculate the fitness function for each individual.

For $1 \leq i \leq M$: Calculate Equation (1) $E(gene(i))$

Select the individual gene(i_{elite}) having the largest fitness with $E(\text{gene}(i_{\text{elite}})) = \min_{1 \leq i \leq M}\{E(\text{gene}(i))\}$.

If the fitness value for the elite individual does not change during $N(>> 0)$ generations, or if the generation number is more than $10N$, exit the loop and terminate. The elite individual is reported as the best matched model.

(III) Otherwise, continue searching by introducing genetic operations of crossover and mutation.

A. Crossover

For $1 \leq h \leq M$: Select one pair of parents gene(i), gene(i') among M individuals randomly and also select a crossover point c randomly.

Let

childgene(h) $= (l_1(i), \cdots, l_c(i), l_{c+1}(i'), \cdots, l_R(i'))$ and compute a fitness value $E(\text{childgene}(h))$.

Select a set of M individuals having higher fitness from childgene(h) and gene(i) $(1 \leq i, h \leq M)$, and update gene(i) using these. Sort gene(i) in higher fitness order.

B. Mutation

For $M/2 \leq i \leq M$: In case $i = m \bmod 4$ $(0 \leq m \leq 3)$, Mutation $m + 1$ is applied to $gene(i)$ with the probability m_0.

(IV) Return to Step (II)

3. ANALYSIS

In this section, we discuss more detail about the first and second steps, using the following simple example (Figure 1) of contour grouping.

The ideal segmentation is that the circle and the two lines are separated into three different layers in Figure 5.

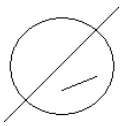


Figure 1: A simple example.

Local Interaction and global inhibition

Consider the contour grouping for edge features using our MGA model.

We assume each local edge feature g_r has information of the position $n(r) = (n_1(r), n_2(r))$ and the local orientation, a(r) [radian]. This pairwise local interaction $f_{g_r g_{r'}}$ between edge features is depicted in Figure 2. This interaction field is similar to the cocircular interaction employed in [1]. In [1], Wersing, Steil, & Ritter used only coarse images, i.e., they assumed that only coarsely sampled features are active in numerical

experiments. Therefore, their interaction field is relatively wide and is not suitable for our cases.

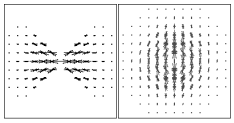


Figure 2: Local interaction for contour grouping; edges $g_{r'}$ in the left picture have $f_{g_r g_{r'}} > 0$ and ones in the right picture have $f_{g_r g_{r'}} < 0$, where g_r is the horizontal edge at the center. Length of each edge codes interaction strength.

Without any global inhibition, the energy value of the ideal segmentation is equal to one where the circle and the short line are together in the same layer and the long line is in another one, or where the two lines are together in the same layer and the circle is in another one. The global inhibition is necessary to separate clusters which should be on different output layers from each other.

Discussion of the first step

We consider that features g_r with edge elements placed on R pixels make contours.

Edges on the circle share positive local interactions with each other in a local neighborhood range as do edges on each line. On the other hand, near the two crossing points of the circle and the long line, the local interaction is negative between an edge on the circle and the one on the line.

At the first step, we prepare M initial individuals for GA, using only the local interactions. In this example, we put $M = 8$ as the number of individuals and $L = 5$ for the output layers. As shown in Figure 3, the process of preparing one individual is as follows.

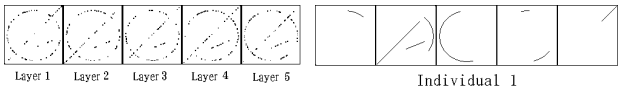
- Assign each feature to one of the layers randomly. (a) shows one example.
- By applying (2) and (3) repeatedly until E_{local} (5) reaches its local minimum, features having positive local interactions with each other tend to group in the same layer. (b) shows how the assignment has changed from (a). Fragmented contours can be seen in each layer.

Discussion of the second step

The crossover operation of step (III) A is needed to create children having better values of the fitness function. The mutations of step (III) B are needed for escape from its local minimums. In our examples, we set a mutation probability to be $m_0 = 40\%$.

An example of the first generation is shown in Figure 4.

At the 29th generation, we get the ideal segmentation in Figure 5.



(a) (b)

Figure 3: The preparation of one individual for GA. (a) One example of a random assignment of features to the output layers. In each output layer, which is represented by one square box, features assigned to the layer appear. After applying the first step, we get the image in Figure(b). Contours become salient but fragmented since E_{local} reaches not its global minimum but only its local one. Because the global inhibition is not considered, we see different groups' contours in the same layer.

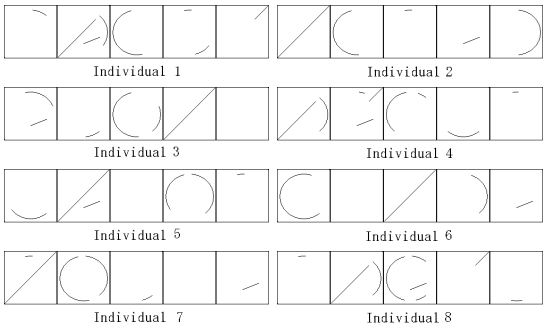


Figure 4: An example of the first generation for the simple example in Figure 1.

Comparison of MGA, the standard GA and CLM

In general, results of GA are very sensitive to selection of a group of initial individuals.

The graph (Figure 6) shows the fitness value of the elite individual in each generation, for MGA, MGA without the first step, and GA (MGA without the first step and special mutations). From the graphs in Figure 6, we see that the convergence speed of MGA without the first step is slower than that of MGA. For MGA without the first step using 8 individuals, we get the ideal segmentation at 139th generation.

The standard method of GA without the first step starts from random initial individuals as in Figure 3(a) and it takes a large number of iterations to get convergence and obtain a proper result. For the method without the first step, it seems to be useless to increase the number of individuals, which probably comes from the fact that the method depends almost entirely on mutations to bind features on the same contour.

In our method, the first step selects the first generation initially by the gradient method, and contributes to grouping some of features having positive local interactions with each other. In the graph (Figure 6), the elite's fitness value of the first generation for MGA has already reached a better value than those for the other methods. Therefore the first step makes MGA

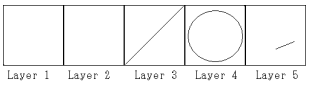


Figure 5: The ideal segmentation of the image in Figure 1; this segmentation is obtained by the elite individual in the 29th generation, starting from the first generation in Figure 4.

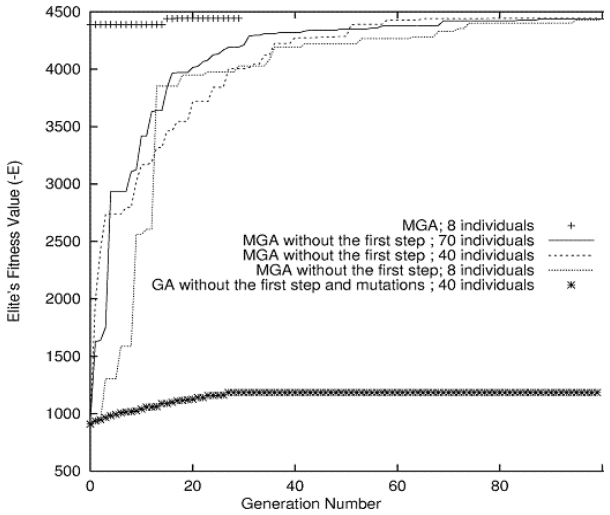


Figure 6: Graphs of the fitness value of the elite individual in each generation against the generation number, for the example in Figure 1. x-axis shows generation number. y-axis shows the negative of the energy function in (1). These graphs are results for MGA, MGA without the first step, and GA.

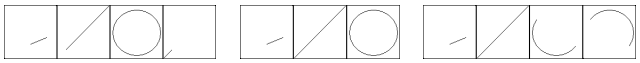
more efficient than the standard GA.

Figure 6 also indicates that the special mutations are effective since even MGA without the first step has the ability to obtain the ideal segmentation, while MGA without the mutations and the first step using 40 individuals is soon trapped into a local minimum state. MGA without the first step has succeeded to group features for this simple example. However, the method does not achieve ideal segmentation for more complicated examples in the next section.

Next we compare MGA grouping result with CLM. The main problem of CLM is that fragmented groupings can occur, due to fast expression of modes. Wersing, Steil & Ritter try to suppress these fragmenting modes using the mechanism of self-inhibitory annealing. However it is not enough because results sensitively change depending on an annealing coefficient (Figure 7). The ideal segmentation is achieved only with an annealing coefficient of 0.9994.

4. OTHER SIMULATIONS

In this section, we apply MGA to more complex



(a)$T_i = 0.9993T_{i-1}$ (b)$T_i = 0.9994T_{i-1}$ (c)$T_i = 0.9995T_{i-1}$

Figure 7: The grouping result of the image in Figure 1 using CLM; since the mechanism of self-inhibitory annealing is not enough to suppress fragmenting modes, a proper grouping does not appear for some coefficients T of self-inhibitory annealing. At each step, T is changed using (a) $T_i = 0.9993T_{i-1}$, (b) $T_i = 0.9994T_{i-1}$ and (c) $T_i = 0.9995T_{i-1}$.

images including some real images. We employ the stopping point parameter $N = 100$.

For noise removal, we use values of $\sum_{r'} \max(0, f_{g_r g_{r'}})$ to determine whether features g_r are noise or not. In the algorithm, if $\sum_{r'} \max(0, f_{g_r g_{r'}})$ is less than a small positive value EPS, we delete the features g_r, since these have small positive interactions with the others. We also delete features with small grayness intensity. We use the limit intensity as 40% of maximum intensity on the image.

For realistic data (Figures 8, 9) we use Prewitt method to detect edges and choose only one edge feature having maximum intensity grayness in each 5×5 pixel square, which is one of squares obtained by subdividing the image.

Figure 10 shows the results obtained using CLM for Figure 9 with coefficients of 0.9994 and 0.9999 for self-inhibitory annealing. These figures show that the results depend critically on these parameters and that fragmented modes and wrong feature bindings appear.

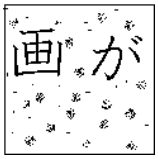
5. DISCUSSION AND CONCLUSIONS

In this paper, the modified genetic algorithm (MGA) is introduced for image segmentation based on the feature clustering problem using an energy function. We propose a new method for creating initial individuals as the first step of our algorithm. At the second step, we apply the genetic algorithm (GA) with the special mutations to the initial individuals generated in the first step. A comparison between the standard method of GA and MGA has been made, and we have demonstrated that the first step and special mutations make the algorithm efficient.

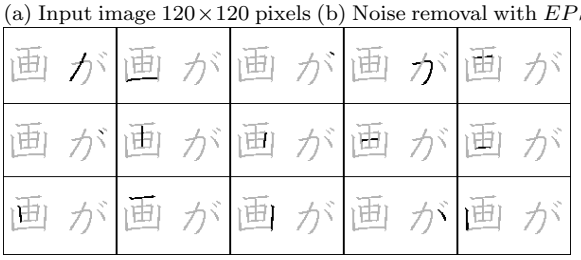
CLM method [1] for feature clustering problem used the annealing method for escaping from the local minimum, which sometimes does not work properly. On the other hand, MGA has succeeded in avoiding this problem and it removes the necessity of parameter adjustments required in CLM.

Some experiments (not reported here because of space limitations) showed that the convergence speed is faster, if the number of individuals is increased.

However the larger number of individuals requires a longer computation time. A parallel processing computer could be used to reduce computation time.



(a) Input image 120×120 pixels (b) Noise removal with $EPS = 6$



(c) Result; main output layers

Figure 8: Grouping of image data. (b) The figure consists of 651 features. (c) 30 output layers and 30 individuals are prepared. At the 100th generation, the proper grouping is obtained. Processing time is 3 minutes. The gray curves show the original image with the noise removed.

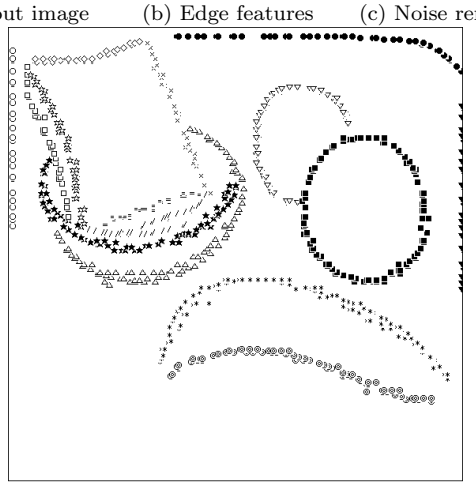
References

[1] H. Wersing, J. J. Steil and H. Ritter, "A competitive-layer model for feature binding and sensory segmentation," *Neural Computation,* vol. 13, pp. 357-387, 2001.

[2] C. Peterson and B. Soderberg, "A new method for mapping optimization problems onto neural networks," *Int. J. Neural Systems,* vol. 1, no. 1, pp. 3-22, 1989.

[3] P. Andrey & P. Tarroux, "Unsupervised segmentation of Markov random field modeled textured images using selectionist relaxation," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 20, no. 3, pp. 252–262, 1998.

(a) Input image (b) Edge features (c) Noise removal



(d) Result; main output layers

Figure 9: (a) 288 × 288 pixels. (c) Image after noise removal with $EPS = 3$; the figure consists of 801 features. (d) 30 output layers and 40 individuals are prepared. Different symbols denote activity in different output layers. At the 599th generation, the proper grouping is obtained. Processing time is 9 minutes.

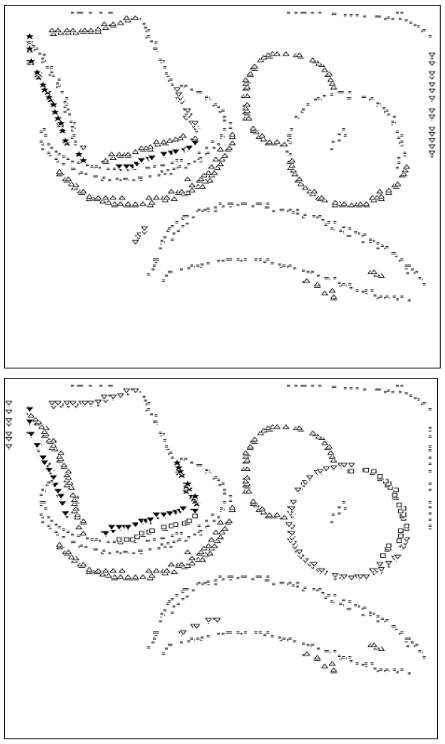


Figure 10: CLM method using (a) $T_i = 0.9994T_{i-1}$. and (b) 0.9999.