An Edge Detection by Using Self-Organization

Hironori Nagai , Yoshikazu Miyanaga , Koji Tochinai

Graduate School of Engineering, Hokkaido University
N13, W8, Kita-ku, Sapporo, 060 Japan

Abstract

This paper proposes a self-organized edge detection. In this method, several clusters are yielded and self-organized according to a gray scale level and the location of pixels. In addition, the comparison among these clusters results in estimated edge. However, after self-organization, clusters are classified into some group according to their properties. In this report, the method which represents the detail distribution of each cluster is introduced. In addition, by using this method, it is shown that the proposed detection of edges can improve the accuracy in some experiments.

1 Introduction

In order to recognize and analyze an object in a digital image, it is important to detect edges from a given image. Therefore many methods to detect edges have been proposed[3][4].

We have proposed a method[5] to detect edges by using self-organized clustering[1][2]. The pixels of an image are clustered by this clustering technique according to a gray scale value and the location of pixel. The comparison of these estimated clusters results in estimated edges. However, the clusters have some properties. On such clusters whose natures is not same, by only the evaluation of the average of the gray scale value, a part of sophisticated difference in actual edges can not be estimated. For example, let us think the area in which a gray scale value is changed quite smoothly. The mean values of clusters which contact each other are not same. Such clusters do not include a part of edges. However, when the smooth edge are tried to detect, non-real edges can be detected there. Therefore, in order to improve the accuracy of edge detection, it is important to explore the detail distribution of each cluster.

In this report, we show what nature each cluster can have, and the method which classify clusters into some groups according to their nature. In order to divide clusters, a variance on the gray scale value and a multiple regression analysis are used. In addition, we show that the accuracy of edge detection can be improved by using this method in some experiments.

2 Self-Organized Clustering

2.1 The Outline of Self-Organization

The self-organizing technique is used to classify input data. This has some cluster nodes. Each cluster has the inner information. They are used on the calculation of a distance and updated adaptively. The whole system is described in Fig.1.

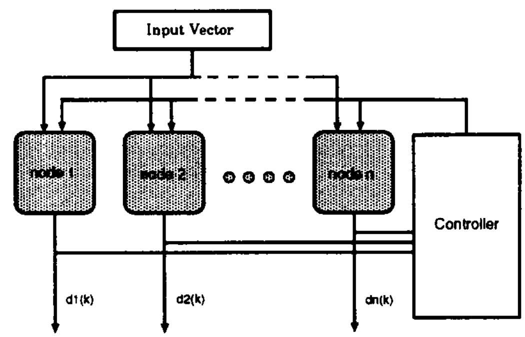

Fig 1: Self-Organized Clustering

2.2 The Division of Pixels

We define the k-th input vector as

$$\boldsymbol{n}(k) = [n(k) \quad x \quad y] \tag{1}$$

where $n(k)$ is the gray scale level and x, y is the location of pixel. The noise can be absorbed within a certain degree by using a 3-dimensional vector. When an input vector is given, the distance is calculated in each node. The input vector is classified into a node whose distance is minimum. The inner information of its node is updated at the time that it is decided to be a member of a certain node. If $\boldsymbol{n}(k)$ is a member of a certain node, $\delta_i(k) = 1$. Otherwise, $\delta_i(k) = 0$.

The internal information of each node consists of numbers of members $s(k)$ and $t(k)$, a centroid of the node $\boldsymbol{m}(k)$. The value $s(k)$ is the number of members on the gray scale values of a pixel, on the other side $t(k)$ is on its location. They are defined as

$$u_i(k) = \sum_{j=0}^{k} \delta_i(j) \tag{2}$$

$$s_i(k) = \sum_{j=0}^{k} \lambda_s^{u_i(k)-u_i(j)} \delta_i(j) \tag{3}$$

$$t_i(k) = \sum_{j=0}^{k} \lambda_t^{u_i(k)-u_i(j)} \delta_i(j) \tag{4}$$

$$m_{i,l}(k) = \begin{cases} \dfrac{1}{s_i(k)} \sum_{j=0}^{k} \lambda_s^{u_i(k)-u_i(j)} \delta_i(j) n_l(j) & : \quad l = 1 \\ \dfrac{1}{t_i(k)} \sum_{j=0}^{k} \lambda_t^{u_i(k)-u_i(j)} \delta_i(j) n_l(j) & : \quad l = 2,3 \end{cases} \tag{5}$$

where l denotes l-th line element of a vector, $l = 1, 2, 3$. The value λ_s, λ_t are weighting coefficients, $0 < \lambda_s < 1, 0 < \lambda_t < 1$ and $\lambda_s \approx 1, \lambda_t \approx 1$.

The definition of the standard Euclidean distance is given as

$$e(k|k-1) = n(k) - m\ (k-1) \tag{6}$$

$$d(n, m) = \sqrt{e(k|k-1) \cdot e(k|k-1)^T} \tag{7}$$

where T denotes a transpose. When a vector is inputed, the distance is calculated in each node from (6),(7). It is classified into a node whose distance is minimum. However, if all distances are more than a certain threshold, a new node is generated. Initial values are $t(1) = 1$, $m_i(1)$ is $n(j)$ which is classified into i-th node first.

Once an input vector is clustered by using this method, the data is used to update the information of a selected node. The inner information is updated as follows:

$$m_{i,l}(k+1) = \begin{cases} \dfrac{s_i(k)}{s_i(k+1)} \lambda_s^{\delta_i(k+1)} m_{i,l}(k) + \dfrac{\delta_i(k+1)}{s_i(k+1)} n_l(k+1) & : \quad l = 1 \\ \dfrac{t_i(k)}{t_i(k+1)} \lambda_t^{\delta_i(k+1)} m_{i,l}(k) + \dfrac{\delta_i(k+1)}{t_i(k+1)} n_l(k+1) & : \quad l = 2,3 \end{cases} \tag{8}$$

$$s_i(k+1) = \lambda_s^{\delta_i(k+1)} s_i(k) + \delta_i(k+1) \tag{9}$$

$$t_i(k+1) = \lambda_t^{\delta_i(k+1)} t_i(k) + \delta_i(k+1) \tag{10}$$

2.3 The Fusion of Nodes

When the analysis of clusters is punished, each node can approach each other. Thus, it is necessary to fuse such clusters. This brings the decrease of calculation quantity and the possibility to generate a new node in new extent. The procedure is shown as follow:
When the two clusters to fuse each other are

$$C_a : m_a(k), s_a(k), t_a(k) \tag{11}$$

$$C_b : m_b(k), s_b(k), t_b(k) \tag{12}$$

and a new node by the fusion is

$$C_c : m_c(k), s_c(k), t_c(k), \tag{13}$$

its node information is calculated as follow:

$$s_c(k) = s_a(k) + s_b(k) \tag{14}$$

$$t_c(k) = t_a(k) + t_b(k) \tag{15}$$

$$m_{c,l}(k) = \begin{cases} \dfrac{s_a(k)}{s_c(k)} m_{a,l}(k) + \dfrac{s_b(k)}{s_c(k)} m_{b,l}(k) & : \quad l = 1 \\ \dfrac{t_a(k)}{t_c(k)} m_{a,l}(k) + \dfrac{t_b(k)}{t_c(k)} m_{b,l}(k) & : \quad l = 2, 3. \end{cases} \tag{16}$$

3 The Former Method of Edge Detection

After self-organization, each cluster is explored whether it includes a part of edges. The method which we have proposed calculates the mean of gray scale value in each node cluster and compared with neighborhood clusters. The criterion on this method is defined as follows:

$$ave(i) = \frac{sum(i)}{mem(i)} \tag{17}$$

where $sum(i)$ denotes the total of gray scale value and $mem(i)$ denotes the total number of members in the i-th node. For the judgement of the edge detection, a square error is used. This expression denotes as follows:

$$se(i, j) = (ave(i) - ave(j))^2 \tag{18}$$

However, in order to improve the accuracy of edge detection, it is necessary to analyze clusters after self-organization. Therefore, we show the method to analyze clusters in the next section.

4 The New Method of Edge Detection

4.1 The Nature of Cluster

After the division of pixels, the clusters have some properties according to their members. They are considered as

- **Group1:** a group of the clusters to which the pixels of the almost uniform gray scale value belong.
- **Group2:** a group of the clusters to which the pixels with additive noise belong.
- **Group3:** a group of the clusters to which the pixels with changed gray scale values smoothly belong.

When such clusters which have different nature are compared with only their average of gray scale values, edge detection is apparently mistaken. Therefore, in order not to mistake edges detection, it is important to analyze clusters.

4.2 Division of Clusters

4.2.1 Variance on the Gray Scale Value

For the method to distinguish the clusters into 3 groups, Group1 and Group2, Group3, which are denoted in 4.1. the variances are used at first. It is used to distinguish Group1 from Group2 and Group3. If its values of a certain cluster is small, its cluster is estimated to be Group1. Otherwise, it is estimated to be Group2 or Group3.

4.2.2 The First Dimensional Approximation by Using Multiple Regression Analysis

In order to distinguish Group2 from Group3, a multiple regression analysis is used. On Group3, it is necessary to explore what direction the gray scale value of pixel changes smoothly to. In order to judge such tendency, each cluster is expressed by the 1-dimensional approximation. If it is considered the pixels which belong to i-th node, it is defined as follow:

$$S_{xy} = a_i x_{ij} + b_i y_{ij} + c_i \tag{19}$$

where S_{xy} is the estimated gray scale value and x, y is the location of pixel, i is the number of node and j is the number of pixels which belong to the i-th node. Therefore, their direction is determined by the values a_i, b_i. Thus, the coefficients a_i and b_i are found from solving the following simultaneous equations

$$\begin{cases} w_{xx}a_i + w_{xy}b_i = w_{xs} \\ w_{xy}a_i + w_{yy}b_i = w_{ys} \\ c_i = \bar{s}_i - (\bar{x}_i a_i + \bar{y}_i b_i) \end{cases} \tag{20}$$

where w_{xy} is covariance between x and y, s is real gray scale value, $\bar{s}$ is the mean value of s.

4.3 The Way of Edge Detection

After self-organization, each node is divided into three groups, Group1 and Group2, Group3. It is explored whether it includes a part of edges among nodes from its analysis. When the neighbor clusters are not Group3 each other, its judgement is determined by (18). Otherwise, it is determined by the results of cluster division and (18). In other words, if the directions of their slopes are same and the square error values is small, a real edge does not exist. However, if their slopes are same and it becomes large, the real edge exists as Fig2. Thus, the cluster in Group3 can have the slope of 8 directions which denotes in Fig3. In Fig3, the combination as a and b,or b and c and so on, are considered to be same direction. In other words, in Fig3 the directions which adjoin each other are considered to be same.

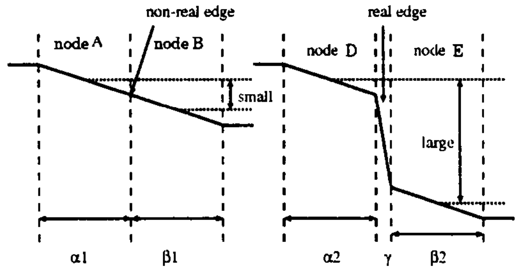

Fig 2: Sectional of an image

5 Examination and Discussion

5.1 The Way of Input Data and Evaluation of Parameters

5.1.1 The Way of Input Data

A way to input a data to this system is quite important on the generation of nodes. In order to prevent nodes issue recklessly, the data had to be input regularly on the element of pixel's location. Therefore, the way is defined in Fig4.

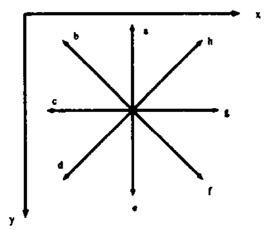

Fig 3: Directions of slope

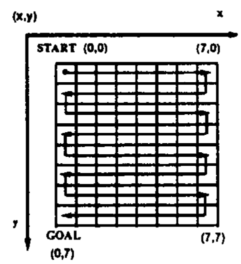

Fig 4: Way of input data

5.1.2 The Establishment of Parameters

When a data is inputed, the distances between all nodes are calculated. If all distances are more than T_s, a new node is generated. If it is set up as large, the generate of node is controlled lower. However, smooth edges can be absorbed and not be detected. On the other hand, if it is set up as small, that indicates the increase of node and calculation quantity. From many other examinations, its value is set up to $T_s = 20$.

The values λ_s, λ_t are forgetting factors. From the nature of input data, they are proper to be set up to $\lambda_s \approx 1, \lambda_t \approx 1$. The value λ_s is a forgetting factor on the gray scale value of pixel and λ_t is one of its location. The λ_s had to be $\lambda_s = 1$ because of the preservation of its information. On the other hand, when the λ_t is set up as small, the generation of nodes is controlled lower by the increase of pursuit on the node pattern. However, in this case, since the node pattern always approaches input data, the noise cannot be absorbed in the narrow area of an image. From many other examinations, it is proper to be $\lambda_t = 0.98$.

When a certain data have been inputed, the distances between nodes are calculated. If their distances are less than T_y, they are mixed. If it is set up as small, few nodes are fused. On the other hand, if it is set up as large, a lot of nodes are fused and the quantity of calculation decreases. However, clusters which include a part of edges can be fused. From many other examinations, it is proper to be $T_y = 15$. From this results, once nodes are generated and after that the fusion of nodes can bring the control of node increase. Thus, in this examinations, the fusion of nodes is performed every 4000 input data.

5.2 Division of Clusters

When the members of cluster can be quite few after self-organization, the results of analysis lack in confidence and the simultaneous equations (20) can not be solved. Such a cluster is almost edge itself. Therefore, they which consist of the under ten members are assumed to be edges and not analyzed.

5.2.1 Analysis of Variance

In order to explore the distribution of a gray scale value, the variance of each cluster was estimated. The result is shown in Fig5. They are rich in variety. Since the clusters in Group3 consist of almost equal gray scale value they are small. Therefore, if they are less than 20, they are in Group3. In order to confirm the propriety of the threshold, their slope was explored. They were almost small.

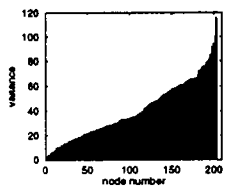

Fig 5: Variance

5.2.2 Slope of Cluster

In clusters whose variance is more than 20, a cluster is expressed by (19) and their slopes, a_i and b_i, are calculated. The results are shown in Fig6. They are rich in variety. By this analysis, they are divided into Group2 and Group3. Either a_i or b_i of clusters in Group3 is large because their gray scale values are changed smoothly. On the other hand, one in Group2 is small because of their non-regularity. Therefore, either a_i or b_i is more than 0.3, it is Group3. Otherwise, it is Group2.

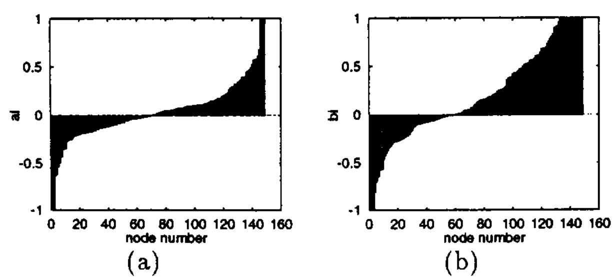

Fig 6: Slopes of clusters(a)a_i,(b)b_i

5.3 Experimental Results

After self-organization, each cluster is analyzed by the method is stated in 4.2. The edges from an image are detected allowing their results. We have showed the detected edge images by the conventional and the new method. In the conventional method, non-real edges have been detected. However, in the proposed method, such mistakes have been overcome. These results have shown that the division of cluster have brought to the accuracy to detect edges from images.

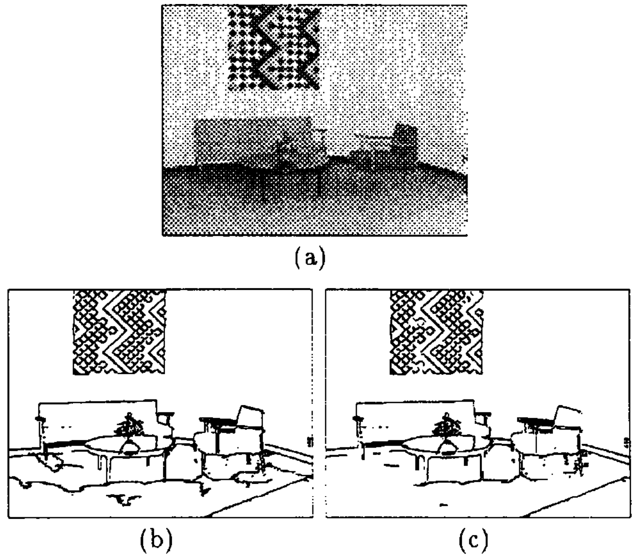

Fig 7: (a)room,(b)Result of the conventional method,(c)Result of the proposed method

6 Conclusion

In this report, we have proposed the method to detect edges from an images by self-organization and divided clusters into 3 groups, Group1 and Group2, Group3 to improve the accuracy of edge detection. In addition, we have showed that the proposed method improves the accuracy of edge detection in some examinations. In the future work, we will study as follow: In this examination, the threshold to divide Group2 from Group3 has been fixed. However, in order to improve the accuracy of edge, they should be set up adaptively, allowing for the nature of their backgrounds.

References

[1] Y.Miyanaga, S.Okumura, K.Tochinai: "On Generalization and Adaptation of a Self-Organized Clustering", IEICE Trans. A, Vol.J75-A, No.7, pp.1207-1215, 1992

[2] Y.Miyanaga, K.Tochinai: "Parallel VLSI Architecture for Multi-Layer Self-Organizing Cellular Network", IEICE Trans. Electron., Vol.E76-C, No.7, pp.1174-1181, 1993

[3] J.Canny,"A computional approach to edge detection",IEEE Trans. Pat. Anal. Machine Intell.,vol.PAMI-9,no4,pp569-698,Nov.1986.

[4] T.Sugiyama, K.Abe: "Edge Feature Analysis by a Vectorized Feature Extractor and Multiple Edges",IEICE Trans. Electron., Vol.J80-D-II, No.6, pp.1379-1389, 1997

[5] H.Nagai, Y.Miyanaga, K.Tochinai: "An Adaptive Edge Detection on the Space Distribution of Images", TECHNICAL REPORT OF IEICE,DSP97-45(1997-06),pp.97-103