RASC2004, pp.122-127,
Nottingham, U.K., December 16-18, 2004

Orientation Selectivity by TAM Network Using Gabor Function Type Receptive Field

Isao Hayashi
Faculty of Informatics, Kansai University
Takatsuki, Osaka 569-1095, Japan
e-mail: ihaya@res.kutc.kansai-u.ac.jp

James R. Williamson
Theater Missile Defense, Lockheed Martin Corp.
MIT, Lexington, MA 02420, U.S.A.
e-mail: jrw@ll.mit.edu

Abstracts: *The TAM (Topographic Attentive Mapping) network with Gabor function type input layer is a biologically-motivated neural network incorporated receptive field. In this paper, we discuss orientation selectivity using Gabor function type receptive field for TAM network.*
Keywords: Neural Network, Vision, Gabor Function, Orientation Selectivity.

1 Introduction

Concerning orientation selectivity of simple cell, complex cell and hypercomplex cell in the human visual cortex, a hypercomplex cell plays important roles to detect contour orientation by slit type field. The slit is a narrow receptive field from 0° to 5° represented by adjacent ON and OFF regions filled, respectively, with plus and minus signs. The sixteen orientations from 0° to 337.5° per 22.5° are detected there. Plenty models imitate the visual cortex have been proposed[1]-[4], and the receptive field models are also in [5]-[7]. Marčelja[6] defined Gabor function as a oscillator which is a complex sinusoidal plane wave of some frequency and orientation. Daugman[7] extended it to two-dimensional format. On the other hand, TAM network is a biologically-motivated neural network[4], [8] and [9]. The TAM network is composed of three layers where feature layer imitates the retina, category layer imitates the lateral geniculate nucleus and in the class layer, the output is given by the name of object grouping.

In this paper, we formulate receptive field type input layer for TAM network using Gabor function and discuss the orientation selectivity of the Gabor function type receptive field. By the Gabor filtering, feature map in the input layer of TAM network is compound from density values at each receptive field for the sixteen orientations. The biological motivation for Gabor filtering to the TAM network lies in constructing like the receptive field in human visual cortex. We formulate here the new TAM network, and show the usefulness of the TAM network through some examples.

2 Gabor Function

Let x and y denotes axis on two-dimensions plane, respectively. The two-dimensional Gabor function, $G(x, y)$, is represented as follows;

$$G(x,y) = Ke^{-\frac{1}{2}(\frac{(x-\mu_x)^2}{\sigma_x^2}+\frac{(y-\mu_y)^2}{\sigma_y^2})} \times \sin(2\pi f_x x\cos\theta + 2\pi f_y y \sin\theta + \phi) \tag{1}$$

where, K is amplitude, (μ_x, μ_y) is the center coordinate of Gabor function, σ_x and σ_y are standard deviations, and f_x and f_y are frequencies. In the case of $\phi = \pi/2$, the sine form is equal to the cosine form of Gabor function.

Gabor filtering is a well known method to detect the contour orientations using the Fourier transform of Gabor function. Given an image, the sixteen orientations of contour in the image are detected. Let the horizontal and vertical scale of the input image denote as R_H and R_V, respectively. In the discrete type Gabor filtering algorithm, the contour image, $C_i(x, y),\ i = 1, 2, \cdots, M,\ 0 \le x \le R_H,\ 0 \le y \le R_V$, for the i-th orientation is calculated by the following convolution process between the Gabor function and the original input image.

$$C_i(x, y) = \sum_{q=1}^{R_V} \sum_{p=1}^{R_H} G_i(x - p, y - q) \times I(p, q) \tag{2}$$

where, G_i is the Gabor function for detecting the i-th orientation, and $I(p, q),\ 0 \le p \le R_H,\ 0 \le q \le R_V$ is the input image.

3 TAM Network with Gabor Function type Receptive Field

The structure of the TAM network is shown in Figure 1. The Gabor filtering process is represented in Figure 2. The Gabor filtering is incorporated in the input layer of the TAM network. First, an image is clipped by the scale of $R_H\ pixel \times R_V\ pixel$. The contour images for sixteen orientations of the input image is next detected at each receptive field by the Gabor filtering. The density from 0 to 255, i.e., black color is 0 and white color is 255 in the PGM format, is then piled as the orientations of the contours. Since a receptive field is here shifting one pixel by one, the center coordinate position of receptive field is equal to a pixel location and the number of receptive fields is totally $R_H \times R_V$. Finally, $R_H \times R_V$ density values are normalized and feature map, f_{ih}, for the i-th feature in the TAM network is constructed as follows;

$$f_{ih} = \frac{\sum_{\{x,y|O_i(x,y)=h\}} C_i(x, y)}{\sum_{y=1}^{R_V} \sum_{x=1}^{R_H} C_i(x, y)}, \quad h = 1, 2, \cdots, L \tag{3}$$

where, $L = 256$.

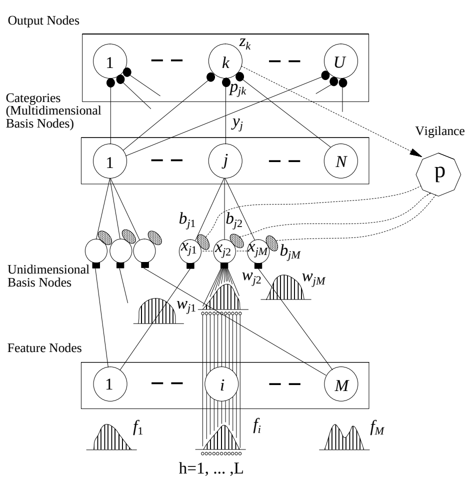

Figure 1: TAM Network

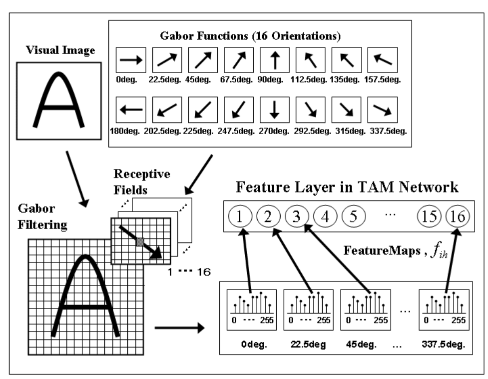

Figure 2: Gabor Filtering Process

After feature maps, $f_{ih},\ i = 1, 2, \cdots, M$, are calculated, the output signal to the category layer, y_j, are calculated using the node's weights, w_{jih}.

$$y_j = \prod_{i=1}^{M} x_{ji} = \prod_{i=1}^{M} \frac{\sum_{h=1}^{L} f_{ih} w_{jih}}{1 + \rho^2 b_{ji}} \tag{4}$$

where x_{ji} are activities, ρ represents the vigilance parameter and b_{ji} are inhibitory weights. Since the sixteen orientations are defined here as the selectivity, we set $M = 16$.

The output prediction, K, is calculated as follows:

$$K = \{k | \max_k z_k\} = \{k | \max_k \sum_{j=1}^{N} y_j p_{jk}\} \tag{5}$$

where, z_k are the output at each node of output layer and p_{jk} are weighted connections.

Let K^* denote the index of the "correct" supervised output class. If the network's output prediction K is not similar enough to K^*, we do $\rho = \rho + \rho^{(step)}$ until either $z_{K^*}/z_K \geq OC$ or $\rho \geq \rho^{(max)}$, where OC is the maximal vigilance level. Once the subject of $z_{K^*}/z_K \geq OC$ is satisfied, the feedback signal y_j^* is calculated, and the learning parameters, w_{jih}, p_{jk}, b_{ji}, are renewed using parameters, α, λ and $b_j^{(rate)}$[4].

The algorithm of the TAM network is represented as follows:

[Step 1] Given an image, the sixteen orientations of contours are detected by Gabor filtering.

[Step 2] The feature map, f_{ih}, is calculated from the density values of the input image.

[Step 3] The output prediction, K, is calculated.

[Step 4] If K is not similar enough to K^*, we do $\rho = \rho + \rho^{(step)}$. When ρ reaches the maximal level, one node is added to categories.

[Step 5] If $z_{K^*}/z_K \geq OC$, the learning step starts. Parameters, w_{jih}, p_{jk} and b_{ji}, are updated.

[Step 6] Until $z_{K^*}/z_K \geq OC$, let the algorithm repeat from step 1 to step 3.

[Step 7] The pruning step starts. The data set in which $f_{si},\ s = 1, 2, \cdots, R$ is divided into learning data and checking data. The information entropy, $H(i) = -\sum_{j=1}^{N} g_j \sum_{k=1}^{U} G_{jk} \log_2 G_{jk}$, is calculated using the learning data for feature selections, where $g_j = \frac{\sum_{s=1}^{R} x_{jis}}{\sum_{j=1}^{N}\sum_{s=1}^{R} x_{jis}}$, $G_{jk} = \frac{\sum_{s \in \psi_k} \gamma_{js} \times p_{jk}}{\sum_{s=1}^{R} \gamma_{js} \times p_{jk}}$, $\gamma_{js} = \prod_{i \in I^*} x_{jis} \times x_{jis}$ and ψ_k is a set of the data of the class k.

[Step 8] The feature, $i^* = \{i | \max_i H(i)\}$, is extracted as an important feature and we set $I^* = \{i^*\}$.

[Step 9] If the condition, $G_{jk} \geq \eta$, is satisfied for checking data at a category j, the link connections between j and outputs $k',\ k' = 1, 2, \cdots, U,\ k' \neq k$, are removed. Simultaneously, the connections between j and features $i' \notin I^*$, are removed, where η is a threshold.

[Step 10] If the condition, $\frac{1}{R}\sum_{s=1}^{R} \gamma_{js} < \theta$, is satisfied for checking data at the category j, the link connections between j and i, and $i' \notin I^*$, are removed, where θ is a threshold.

[Step 11] If the condition, $\varphi_{jK} = \frac{\sum_{s \in \Gamma_K} \gamma_{js} \times p_{jK}}{\sum_{j=1}^{N} \sum_{s \in \Gamma_K} \gamma_{js} \times p_{jK}} \geq \xi$, is satisfied for checking data at K, the link connections between K and categories, $j',\ j' = 1, 2, \cdots, N,\ j' \neq j$, are removed, where ξ is a threshold.

[Step 12] When a category has lost connections to all outputs or features, the category is removed. Any output and feature which has been disconnected from all categories is also removed.

[Step 13] Until all features are selected at step 6, let the algorithm repeat from step 7 to step 12.

When the algorithm is terminated, the neural network whose needless connections and nodes are pruned is obtained. We should notice that the algorithm is a kind of fuzzy tuning methods since the data procedure is the same as that of fuzzy logic. Thus, we can acquire fuzzy rules from the TAM network as a knowledge representation.

4 Character Recognitions

In order to show the efficiency of the receptive field type input layer of the TAM network, some examples are here illustrated. The alphabets are filled in the electronic pad, whose size is 15 $pixels \times$ 15 $pixels$, corresponding to the input image. The normal alphabets as training images are first filtered and translated to feature maps at the input layer. After the learning of the TAM network, the recognition rate for the checking images is calculated. The following three views are here pointed.

1. Concerning alphabets, "A and B", analyzing the orientation selectivity of the input images after Gabor filtering and estimating the recognition rate after the learning of the TAM network.
2. Concerning alphabets, "A and B", comparing robustness of four types of checking images.
3. Concerning alphabets, "A to Z", estimating the recognition rate under reducing some pixels from original alphabets.

Examples of the training images, "A", are shown in Figure 3. The checking images are shown in Figure 7. The contour images of sixteen orientations are shown in Figure 4, which are the left side character 'A' in Figure 3. The orientation of the left-upper side indicates 0° and be moving the right side one by one to be 22.5°, 45°, $\cdots$, 337.5°, respectively. We set here the parameters of Gabor function as $\sigma_x = 1.99$, $\sigma_y = 1.92$, $f_x = f_y = 0.127$, and $K = 1.0$. The contours according to each orientation are clearly detected. The contour images of the orientation, 135.0°, for changing σ_x are shown in Figure 5. The Gabor functions are then shown in Figure 6. The value of σ_x of the left side image is 0.5 and be moving the right side one by one to be 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, respectively. Other parameters except for σ_x are set as $\sigma_y = 1.92$, $f_x = f_y = 0.127$, and $K = 1.0$. The larger σ_x is, the more vague the contour image is. The orientation selectivity for changing σ_y and the amplitude, K, were sensitive as well as σ_x. The frequency, f_x and f_y, were not so sensible to orientation selectivity.

Next, the recognition rates of "A", are shown in Figure 8. The larger σ_x is, the lower the recognition rate of the checking image is. The recognition rate of the training image is constantly keeping in high. From Figure 5 and 8, we found that the contour orientation can be sharply detected after adjusted the optimal value of σ_x at relatively small region. That is why the narrow Gabor function can detect sharply the contour orientation since the value of σ_x means the width of Gabor function. On the other hand, the large σ_x can detect the orientations of the whole image, but the contour becomes obscure.

The robustness of the TAM network is estimated using four types of the checking images, "A and B". Figure 9 shows the recognition rates for four types of checking images, 'A', which are shifting characters, rotations of characters, adding noise to characters, and reducing pixels in characters as divided in Figure 7. The recognition rates of shifting and rotation of characters among four types are almost higher than 70.0%. The larger σ_x is, the lower the recognition rate of characters added noise is.

In order to discuss the usefulness of TAM network, the recoginition rates of "A to Z" are respectively compared under reducing 20% pixels, 30% pixels and no reducing from original

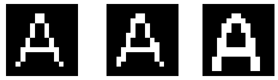

Figure 3: Training Image of ‘A’

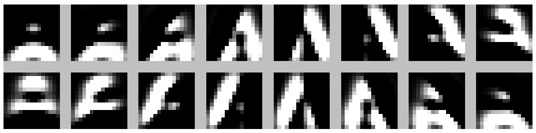

Figure 4: Image after Gabor Filtering

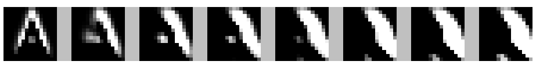

Figure 5: Contours of Input Images

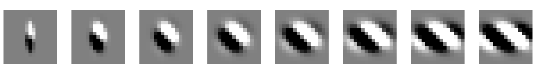

Figure 6: Gabor Functions

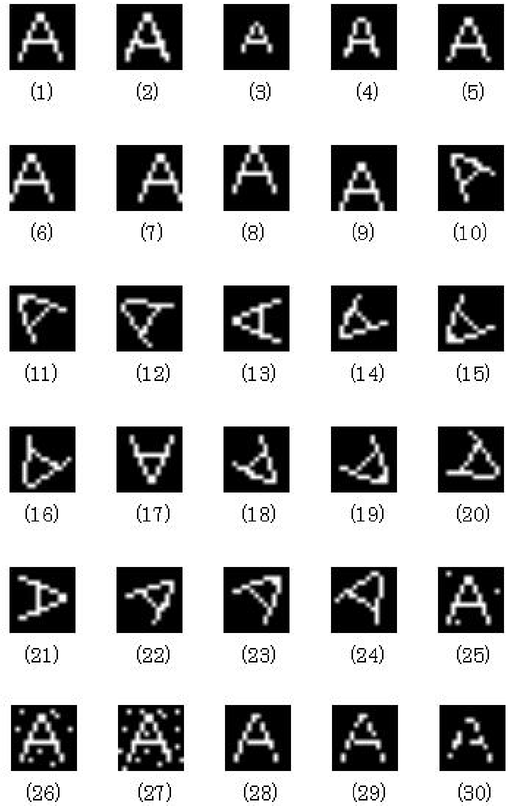

Figure 7: Checking Image of ‘A’

characters. Figure 10 shows the result of recognition rate representing the two-dimensional plane of both of reducing 20% pixels and 30% pixels. The recognition rates are the average values of 15 different types of patterns under reducing pixels for the same character, and the experiments at each type of pattern are conducted 30 times under changing the order of training data. The average recognition rate of reducing 20% pixels is 26.6% and 12.8% for reducing 30% pixels comparing with 69.7% for no reducing pixels. The alphabets, ‘I’, ‘J’, ‘K’, ‘L’, ‘T’, ‘Y’ and ‘Z’, are relatively robust for reducing pixels since there are no similar characters and they are drawn with a few strokes. Examples of reducing 30% pixels are shown in Figure 11. The alphabets, ‘C’, ‘G’, ‘O’, ‘Q’ ‘N’, ‘M’, ‘H’, ‘B’, ‘P’ and ‘R’ are weak of robustness since ‘C’, ’G’, ’O’ and ‘Q’ are similar each other, ‘N’, ‘M’ and ‘H’ are also similar each other, and ‘P’ and ‘B’ are similar to ‘R’, as shown in Figure 11. The recognition rates of the alphabets, ‘C’, ‘S’, ‘V’ and ‘X’ are depending to the type of reducing pixels since they are including the important pixels inside for recognize itself. Examples are also shown in Figure 11.

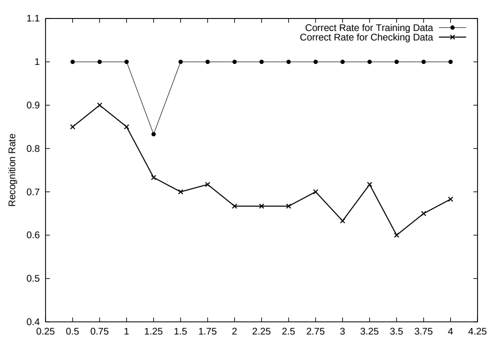

Figure 8: RR under Changing σ_x

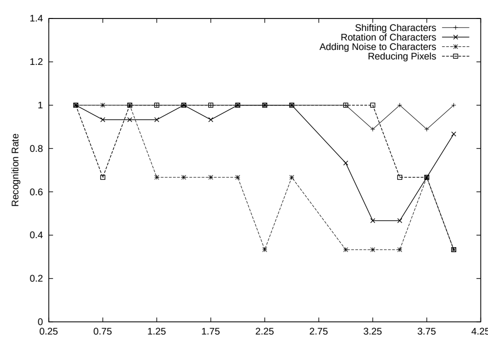

Figure 9: RR of Four Types of Image ‘A’

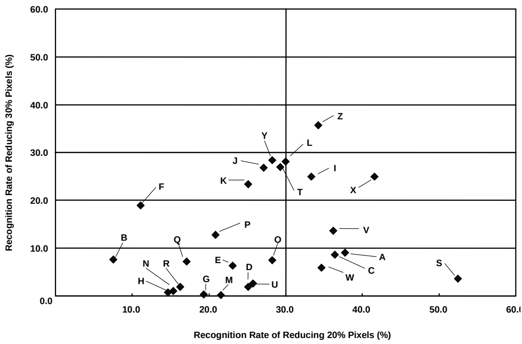

Figure 10: RR of Alphabets under Reducing Pixels

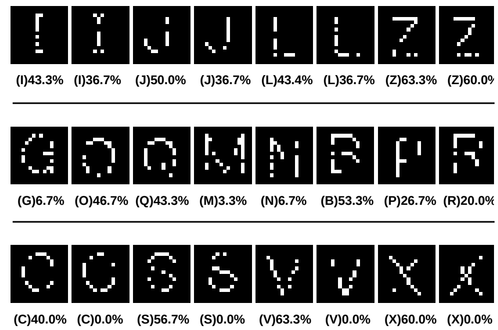

Figure 11: Examples of Reducing 30% Pixels

5 Conclusions

We formulated here the TAM network with Gabor function type receptive field and discuss its orientation selectivity. This research is partially supported by the Ministry of Education, Culture, Sports, Science and Technology of Japan under Grant-in-Aid for Scientific Research number 14580433.

References

[1] S.Grossberg (1999). How does the cerebral cortex work? Learning, attention, and grouping by the laminar circuits of visual cortex. In *Journal of Spatial Vision*, Vol.12, No.2, pages 163-185, 1999.

[2] H.Neumann and W.Sepp (1999). Recurrent V1-V2 interaction in early visual boundary processing. In *Journal of Biological Cybernetics*, Vol.81, pages 425-444, 1999.

[3] K.Fukushima (2001). Recognition of partly occluded patterns: a neural network model. In *Journal of Biological Cybernetics*, Vol.84, No.4, pages 251-259, 2001.

[4] J.R.Williamson (2001). Self-organization of topographic mixture networks using attentional feedback. In *Journal of Neural Computation*, Vol.13, pages 563-593, 2001.

[5] A.D.Pollen and S.F.Ronner (1983). Visual cortical neurons as localized spatial frequency filters. In *Journal of IEEE Transactions of System, Man and Cybernetics*, Vol.SMC13, pages 907-916, 1983.

[6] S.Marčelja (1980). Mathematical description of the responses of simple cortical cells. In *Journal of Optical Society of America*, Vol.70, No.11, pages 1297-1300, 1980.

[7] J.Daugman (1985). Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. In *Journal of Optical Society of America*, Vol.2, No.7, pages 1160-1169, 1985.

[8] I.Hayashi and J.R.Williamson (2001). Acquisition of fuzzy knowledge from topographic mixture networks with attentional feedback. In *Proceedings of the International Joint Conference on Neural Networks , IJCNN'01*, pages 1386-1391, 2001.

[9] I.Hayashi and H.Maeda (2003). A formulation of fuzzy TAM network with Gabor type receptive fields. In *Proceedings of the 4th International Symposium on Advanced Intelligent Systems, ISIS2003*, pages 620-623, 2003.