Proceedings of the 6th IASTED International Conference
SIGNAL AND IMAGE PROCESSING
August 23-25, 2004, Honolulu, Hawaii, USA

FEATURES EXTRACTION USING A GABOR FILTER FAMILY

Danian Zheng, Yannan Zhao, Jiaxin Wang
State Key Laboratory of Intelligent Technology and Systems
Department of Computer Science and Technology
Tsinghua University, Beijing 100084, China
email: zdn02@mails.tsinghua.edu.cn

ABSTRACT
Gabor filters possess the optimal localization properties in both spatial and frequency domain, and they have been successfully used in many applications. But how to design a set of befitting Gabor filters for a specific application has maybe puzzled many users for a long time. In this paper, we purpose designing a common set of Gabor filters – a Gabor filter family to solve the problem. The Gabor filters in the family are well combined to capture the whole frequency spectrum in all directions. And we can extract many meaningful features using the Gabor filter family. Experimental results in textures and characters demonstrate these features commendably expressing the local information with the different frequencies and orientations in the image. The Gabor filter family designed by us can be also used in some other applications.

KEY WORDS
Features extraction, Gabor filters, Gabor filter family, texture feature, and character feature

1 Introduction

In recent years, Gabor filters have been successfully used in many applications, such as texture segmentation/classification [3,4,11], target detection, character recognition [7,8], fingerprint recognition [5,9], face recognition [6,10], fractal dimension management, document analysis, edge detection [2], image analysis and compression [1]. Gabor filters have received considerable attention, because they possess the optimal localization properties in both spatial and frequency domain.

Designing some befitting Gabor filters for a particular processing task and reducing the computation time are the common difficulties, and they should be well solved in all these applications. In texture processing, some authors consider the design of a single Gabor filter to segment a two-texture image. The output of a Gabor-filtered texture is modeled well by a Rician distribution, and a measure of total output power is used to select the center frequency of filter and to estimate the Rician statistics of the Gabor- filtered image [3]. Other authors utilize the concept of multi-resolution with parameter selection to do texture analysis, and reduce the computation time by diminishing the image size according to several different sets of parameters [4]. In image analysis and compression, Daugman presented a three-layered neural network for transforming two-dimensional discrete signals into ge- neralized non-orthogonal 2-D Gabor representations for image analysis, segmentation and compression. The Gabor filters used in his paper are based on a biologically inspired log-polar ensemble of dilations, rotations and translations of a single underlying 2-D Gabor wavelet template. The neural network is used for finding the optimal coefficients of Gabor transform [1]. In edge detection, some authors investigate a 2-D Gabor odd filter-based detector, and the overall performance of this detector is almost identical to that of the first derivative of Gaussian [2]. But differing from them, we purpose designing a common multi-resolution Gabor filter family used to extract features from the images. In this paper, two issues are mainly considered: one is to design a preset common set of Gabor filters for these applications, and we call them a Gabor filter family; the other is to extract features using the Gabor filter family from an image. After the 1-D Gabor function had been proposed in 1946 by Gabor in his paper ”Theory of communication”, Daugman extended it to a 2-D Gabor filter and showed it providing simultaneous optimal resolution in both the space and frequency domain. The Gabor filters defined in our paper have the equivalent energy 1, and they are all derived from a common basic Gabor filter by rotating and dilating. These Gabor filters are well combined to cover over the whole 2-D frequency domain. The features extracted using Gabor filters represent the local information in the image. We can take features from the amplitude or phase after convolving the complex Gabor filters with the image. In this paper, we give out four feature examples. In order to reduce the computational complexity, we can choose the numbers of Gabor filters (frequency number and orientation number), features and convolving coefficients (decimation).

The remainder of this paper has four parts. In section 2, we give out the function definition of Gabor filter, and discuss the parameters selection, and design three different Gabor filter families. In section 3, four kinds of features extracted using a Gabor filter family are presented. In section 4, we do three experiments: experiment A shows the features on the images consisting of different frequency contents, experiments B and C show the features on texture and character images. In the last section, we give an overview to the features and the Gabor filter family.

2 Designs for Gabor Filter Family

2.1 Gabor Filter Family

The general functional form of a 2-D Gabor filter family can be specified in eq.(1) and eq.(2), in terms of spatial domain impulse response and its frequency domain response:

$$h(x,y;f,\theta) = \frac{1}{\sqrt{\pi\sigma_1\sigma_2}}\exp\left(-\frac{1}{2}\left(\frac{R_1^2}{\sigma_1^2}+\frac{R_2^2}{\sigma_2^2}\right)\right)\cdot\exp\left(i(f_x x+f_y y)\right) \quad (1)$$

where $R_1 = x\cos\theta + y\sin\theta$, $R_2 = -x\sin\theta + y\cos\theta$, $\sigma_1 = \frac{c_1}{f}$, $\sigma_2 = \frac{c_2}{f}$, $f_x = f\cos\theta$, $f_y = f\sin\theta$, c_1 and c_2 are two constants.

The coefficient $\sqrt{\pi\sigma_1\sigma_2}$ can guarantee that the energies of different Gabor filters in the family are all equivalent to 1, i.e. $\|h\|^2 = \int\int hh^* dxdy = 1$.

$$H(u,v;f,\theta) = 2\sqrt{\pi\sigma_1\sigma_2}\exp\left(-\frac{1}{2}(\sigma_1^2(S_1-f)^2+\sigma_2^2 S_2^2)\right) \quad (2)$$

where $S_1 = u\cos\theta + v\sin\theta$, and $S_2 = -u\sin\theta + v\cos\theta$.

Gabor filters are spatial sinusoids localized by a Gaussian window, and they are orientation and frequency sensitive band pass filters. In eq.(1), x and y are the digital pixel ordinates in the image. The parameters σ_1 and σ_2 are the standard deviations of 2-D Gaussian envelope. The central frequency of the pass band is f, and the spatial orientation is θ.

If let $\sigma_1 = \sigma_2 = \sigma$, then eq.(1) and eq.(2) can be simplified as eq.(3) and eq.(4)

$$h(x,y;f,\theta) = \frac{1}{\sqrt{\pi}\sigma}\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)\cdot\exp\left(i(f_x x+f_y y)\right) \quad (3)$$

$$H(u,v;f,\theta) = 2\sqrt{\pi}\sigma\exp\left(-\frac{\sigma^2((u-f_x)^2+(v-f_y)^2)}{2}\right). \quad (4)$$

The complex Gabor filter comprises two components. One is the real part: $\frac{1}{\sqrt{\pi\sigma_1\sigma_2}}\exp\left(-\frac{1}{2}\left(\frac{R_1^2}{\sigma_1^2}+\frac{R_2^2}{\sigma_2^2}\right)\right)\cos(fR_1)$, and the other is the imaginary part: $\frac{1}{\sqrt{\pi\sigma_1\sigma_2}}\exp\left(-\frac{1}{2}\left(\frac{R_1^2}{\sigma_1^2}+\frac{R_2^2}{\sigma_2^2}\right)\right)\sin(fR_2)$.

The real part is even symmetry, whereas the imaginary part is odd symmetry. For example, fig.1 shows the real part and imaginary part of a Gabor filter, where $f = \frac{\pi}{3}$, $\theta = 0$, and $c_1 = c_2 = \pi$.

The frequency response of another Gabor filter is shown in fig.2, where $f = \frac{\pi}{3}$, $\theta = \frac{\pi}{4}$, and $c_1 = c_2 = \pi$. The frequency center in the (u,v) plane is (f_x, f_y), where $fx = f\cos\theta = 0.7405$ and $f_y = f\sin\theta = 0.7405$. And this indicates that a Gabor filter is a Gaussian filter shifted to the position (f_x, f_y) in frequency.

2.2 Parameters Selection

There are four parameters f, θ, c_1 and c_2 to be selected.

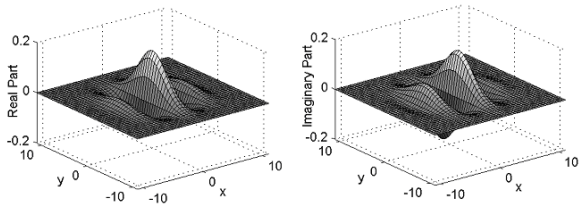


Figure 1. The real part(left) and imaginary part(right) of a Gabor filter.

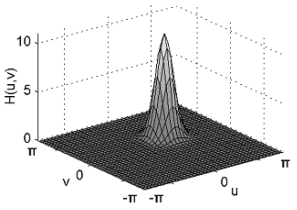


Figure 2. The frequency response of a Gabor filter.

The central frequency of the pass band f can be selected from the interval $[0,\pi]$. A series of frequencies in eq.(5) or eq.(6) is usually used

$$f_k = \frac{\pi}{k+1}, k = 0,1,2,\cdots \quad (5)$$

$$f_k = \frac{\pi}{\alpha^k}, \alpha > 1, k = 0,1,2,\cdots. \quad (6)$$

The sampling in the frequency interval of the former is denser than the latter in general. And in the following of this paper, eq.(6) is used.

The orientations can be distributed uniformly in the interval $[0,\pi]$, such as

$$\theta_m = \frac{m\pi}{M}, m = 0,1,\cdots,M-1. \quad (7)$$

Generally, authors select $M = 4$ in Chinese character recognition, and this just meet the need of the stroke orientations in Chinese characters. And we select the orientation number $M = 8$ in this paper.

Design 1: If $\sigma_1 = \sigma_2 = \sigma$(i.e. $c_1 = c_2 = c$), design an optimal Gabor filter family.

First we'll select the constant c in $\sigma(f_k) = \frac{c}{f_k}$, then we'll choose the α in $f_k = \frac{\pi}{\alpha^k}$.

The frequency response of every Gabor filter with the parameters f_k and θ_m is a 2-D Gauss function as shown in fig.2, and the pass band width is $\frac{2}{\sigma(f_k)}$. In the 2-D frequency domain $(u,v) \in ([-\pi,\pi],[-\pi,\pi])$, The pass band projection is a circle, whose center and diameter are $(f_k\cos\theta_m, f_k\sin\theta_m)$ and $\frac{2}{\sigma(f_k)}$. These circles and frequency responses are shown in fig.3, respectively.

If f_k is fixed and $\theta_m(m = 0,1,\cdots,M-1)$ is changed, the centers of these circles make up of a bigger circle with the center $(0,0)$ and the radius f_k. To make

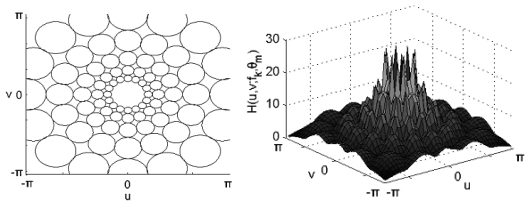
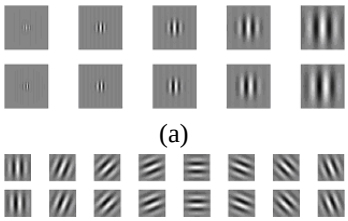


Figure 3. The pass bands and frequency responses of a Gabor filter family, where $\theta_m = \frac{m\pi}{8}$, $f_k = \frac{\pi}{1.4886^k}$ and $\sigma(f_k) = \frac{5.0930}{f_k}$.



(a)

(b)

Figure 4. The real parts and imaginary parts of some members in the above Gabor filter family: (a)different frequencies $f_k = \frac{\pi}{1.4886^k}, k = 0 \sim 4$, (b)different orientations $\theta_m = \frac{m\pi}{8}, m = 0 \sim 7$.

these circles be joined, parameters should satisfy eq.(8).

$$2M \cdot \frac{2}{\sigma(f_k)} \geq 2\pi f_k \tag{8}$$

If θ_m is fixed and $f_k(k = 0, 1, 2, \cdots)$ is changed, the centers of these circles make up of a line. To make these circles be joined, parameters should satisfy eq.(9).

$$\frac{1}{\sigma(f_{k-1})} + \frac{1}{\sigma(f_k)} \geq f_{k-1} - f_k \tag{9}$$

Thus all these circles are joined together and they can cover over the frequency domain. From eq.(8), we can educe $c \leq \frac{2M}{\pi}$. Here $M = 8$, and we select $c = \frac{16}{\pi} \approx 5.0930$. From eq.(9), we can educe $\alpha \leq \frac{c+1}{c-1}, c > 1$. Substituting c with 5.0930, we'll get $\alpha \leq 1.4886$. And we select $\alpha = 1.4886$.

Since all parameters have been selected, we show some members of this Gabor filter family in fig.4. These samples with different frequencies and orientations indicate that each Gabor filter can be got from another one by dilating and rotating. All of them have the same ridge and valley numbers.

Design 2: If $f_k = \frac{\pi}{2^k}, k = 0, 1, 2, \cdots$(i.e. $\alpha = 2$ in eq.(6)), design an optimal Gabor filter family.

We only need to select the parameters c_1 and c_2.

Similarly, we can get the following two inequalities:

$$2M \cdot \frac{2}{\sigma_2(f_k)} \geq 2\pi f_k, \tag{10}$$

and

$$\frac{1}{\sigma_1(f_{k-1})} + \frac{1}{\sigma_1(f_k)} \geq f_{k-1} - f_k. \tag{11}$$

From eq.(10), we can draw $c_2 \leq \frac{2M}{\pi}$, and here we select $c_2 = \frac{16}{\pi} \approx 5.0930$ which is same to design 1. From eq.(11), we can draw $c_1 \leq 3$ and select $c_1 = 3$. Fig.5 and fig.6 show this Gabor filter family and some of its members.

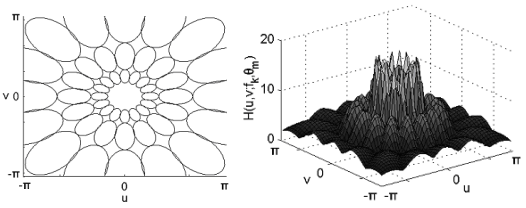
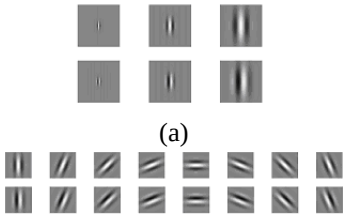


Figure 5. Another Gabor filter family, where $\theta_m = \frac{m\pi}{8}$, $f_k = \frac{\pi}{2^k}$, $\sigma_1(f_k) = \frac{3}{f_k}$ and $\sigma_2(f_k) = \frac{5.0930}{f_k}$.



(a)

(b)

Figure 6. Members with (a)different frequencies $f_k = \frac{\pi}{2^k}, k = 0 \sim 2$, (b)different orientations $\theta_m = \frac{m\pi}{8}, m = 0 \sim 7$.

Comparing the Gabor filter family in design-2 with the one in design-1, we can find that the former can use the fewer number Gabor filters than the latter to cover over the same size area in the 2-D frequency domain.

Design 3: If $f_k = \frac{f_0}{2^k}, k = 0, 1, 2, \cdots$, and $f_0 = \frac{3\pi}{4}$, we can obtain $c_1 = 3$ and $c_2 = 5.0930$ by the same method. Thus we get the third Gabor filter family shown in fig.7.

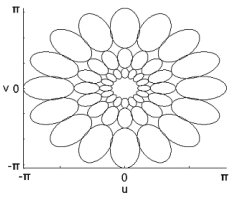


Figure 7. The third Gabor filter family, where $\theta_m = \frac{m\pi}{8}$, $f_k = \frac{3\pi}{4 \cdot 2^k}$, $\sigma_1(f_k) = \frac{3}{f_k}$ and $\sigma_2(f_k) = \frac{5.0930}{f_k}$.

We think the third Gabor filter family is more reasonable than the second one. Because the pass bands of Gabor

filters with $f_0, f_1, f_2, \cdots$ are $[\frac{\pi}{2}, \pi], [\frac{\pi}{4}, \frac{\pi}{2}], [\frac{\pi}{8}, \frac{\pi}{4}], \cdots$, respectively. If the central frequency f is cut down a half, the bandwidth will be cut down a half, too. And the coefficients of the Gabor filter decomposition can be done with a radix-2 decimation.

Only design 3 is used in the rest of this paper.

3 Feature Extraction

The Gabor filter family captures the whole frequency spectrum, both amplitude and phase. For Gabor feature extraction, we convolve the image I with every Gabor filter of the Gabor filter family at every pixel (x, y) as eq.(12)

$$\begin{aligned} &G(x, y; f_k, \theta_m) \\ &= \textstyle\sum_{x\prime} \sum_{y\prime} I(x - x\prime, y - y\prime) h(x\prime, y\prime; f_k, \theta_m), \end{aligned} \tag{12}$$

where $I(x, y)$ is the pixel intensity.

Thus we have got $3 \times 8 = 24$ output images $G(f_k, \theta_m)$, $k = 0 \sim 2$, $m = 0 \sim 7$, and each image has the same size to the image I. We assume the image I has $X \times Y$ pixels. The pass bandwidths corresponding to f_0, f_1 and f_3 are $\frac{\pi}{2}$, $\frac{\pi}{4}$ and $\frac{\pi}{8}$, so $G(f_0, \theta_m)$, $G(f_1, \theta_m)$ and $G(f_2, \theta_m)$ can be done with a radix-2 decimation, a radix-4 decimation and a radix-8 decimation, respectively. And the sizes of $G(f_0, \theta_m)$, $G(f_1, \theta_m)$ and $G(f_2, \theta_m)$ are reduced to $\frac{X}{2} \times \frac{Y}{2}$, $\frac{X}{4} \times \frac{Y}{4}$ and $\frac{X}{8} \times \frac{Y}{8}$. Each new element of $G(f_k, \theta_m)$ after decimation can be the value in the corresponding block center, or the average value in the corresponding block.

Now the following $G(f_k, \theta_m)$ all denotes the output image after decimation.

- The phase information of $G(f_k, \theta_m)$ can be taken as a feature, because it contains information about the edge locations and other details in the image I.

$$F_1(x, y; f_k, \theta_m) = phase(G(x, y; f_k, \theta_m)) \tag{13}$$

- The amplitude of $G(fk, \theta_m)$ can be taken as a feature, and it contains some oriented frequency spectrum in every local of the image I.

$$F_2(x, y; f_k, \theta_m) = |G(x, y; f_k, \theta_m)| \tag{14}$$

- The square sum of the different frequency responses with the same orientation can be taken as a feature, and it denotes the local energy in certain orientation.

$$\begin{aligned} F_3(x, y; \theta_m) \quad &= F_2(x, y; f_0, \theta_m)^2 \\ &+ \tfrac{1}{4} F_2(\lfloor \tfrac{x}{2} \rfloor, \lfloor \tfrac{y}{2} \rfloor; f_1, \theta_m)^2 \\ &+ \tfrac{1}{16} F_2(\lfloor \tfrac{x}{4} \rfloor, \lfloor \tfrac{y}{4} \rfloor; f_2, \theta_m)^2 \end{aligned} \tag{15}$$

In eq.(15), $F_2(f_0, \theta_m)$ is a $\frac{X}{2} \times \frac{Y}{2}$ image, and $F_2(f_1, \theta_m)$ is a $\frac{X}{4} \times \frac{Y}{4}$ image, and $F_2(f_2, \theta_m)$ is a $\frac{X}{8} \times \frac{Y}{8}$ image. The result $F_3(\theta_m)$ is a $\frac{X}{2} \times \frac{Y}{2}$ image, so each pixel of $F_2(f_1, \theta_m)$ is split into 4 pixels and each pixel of $F_2(f_2, \theta_m)$ is split into 16 pixels equally.

- The orientation in which the local has the maximum energy can be taken as a feature, too.

$$\begin{aligned} &F_4(x, y) = k, \\ &where \ F_3(x, y; \theta_k) = \max_{m=0\sim 2} \{F_3(x, y; \theta_m)\}. \end{aligned} \tag{16}$$

There are some other features that can be extracted using the Gabor filter family and not enumerated here. And we can apply one, or several combination of these features in our works.

4 Experiments

Experiment A. Features On Different Frequencies

We have designed two images that contain different frequency components and extracted their features using the Gabor filter family. Fig.8 shows our results.

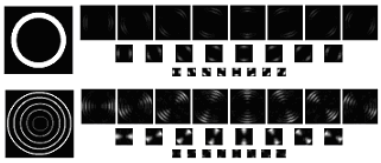


Figure 8. The F_2 features of two images with different frequency components.

Here we only extract out F2 features. The two images are both 64×64 pixels. Features $F_2(f_0, \theta_m)$, $F_2(f_1, \theta_m)$ and $F_2(f_2, \theta_m)$, $m = 0 \sim 7$ are 32×32, 16×16 and 8×8 matrixes, respectively. And they are shown in sequential three layers in fig.8. Comparing these feature images, we can find out two facts: one is that the two images are isotropy and every feature in the same layer contains equal energy; the other is that the energy distributions of the two images are different: the top one is mainly in the bands $[\frac{\pi}{4}, \frac{\pi}{2}]$ and $[\frac{\pi}{8}, \frac{\pi}{4}]$, but the bottom one is mainly in the bands $[\frac{\pi}{2}, \pi]$ and $[\frac{\pi}{4}, \frac{\pi}{2}]$.

Experiment B. Texture Feature

In this experiment, we use two texture images, and extract out their F_2 features like experiment A. Fig.9 shows the two images and their features.

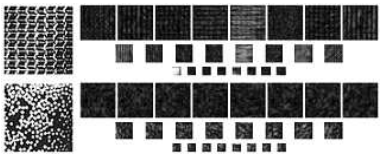


Figure 9. The F_2 features of two texture images.

These features indicate the energy distribution of every image: the top image has a very strong energy at 0 and $\frac{\pi}{2}$ orientations; the bottom has about equal energy at every orientation.

If we want to classify or recognize these textures, we can add up all the square values in every feature image, and then let all the sums make up of a 3×8 vector: $V[8k+m] = \sum_{x,y} F_2(x,y;f_k,\theta_m)^2$, where $k=0\sim2$, $m=0\sim8$.

The vectors produced by the two textures are very unlike, and they can distinguish the two textures very well (shown in fig.10).

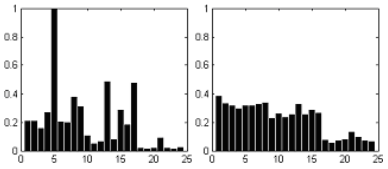


Figure 10. The normalized 8×3 vectors produced by the two texture images.

If we want to segment out these textures from other images, we can first select out some Gabor filters corresponding to those stronger features from the Gabor filter family, then convolve the image with the selected Gabor filters, and then add up all the outputs of the Gabor filters, thus the areas that contain the target texture will have the strongest responses in the image, finally we can segment out the texture areas from the output easily. For example, we can find that the elements V[4] and V[16] in the left vector are much larger than those in the right vector in fig.10, so we use the two relevant Gabor filters to segment out one texture from the mixed-texture image. $Ouput(x,y) = F_2(x,y;f_0,\theta_4)^2 + \frac{1}{16}F_2(\lfloor\frac{x}{4}\rfloor,\lfloor\frac{y}{4}\rfloor;f_2,\theta_0)^2$.

For illuminating the performance of this texture segmentation method, we show the results by using one Gabor filter ($F_2(f_2,\theta_0)$) and multi Gabor filters ($F_2(f_0,\theta_4)$ and $F_2(f_2,\theta_0)$) comparatively in fig.11. The latter result is better.

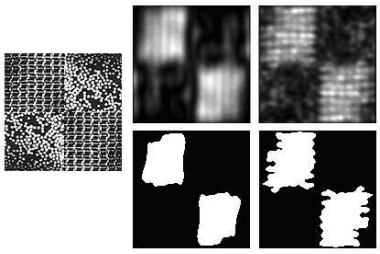


Figure 11. Texture segmentations using one or multi Gabor filters.

Experiment C. Character Feature

In the last experiment, we give out an example on character feature extraction. The following fig.11 shows two character images (letters '0' and 'A', size 64×64) and their $F_2(f_2,\theta_m)$ and F_4 features.

The $F_2(f_2,\theta_m)$ features are all 8×8 matrixes.

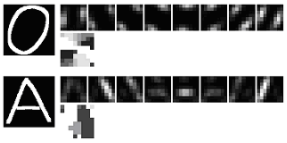


Figure 12. The $F_2(f_2,\theta_m)$ and $F_4(f_2)$ features of two character images.

The $F_4(f_2)$ feature is an 8×8 matrix, because it comes from eight F_3 features on the same image. In the feature image, the strongest brightness denotes the orientation $\frac{7\pi}{8}(k=7)$, and the weakest brightness denotes the orientation $0(k=0)$. This feature contains the information of the stroke orientations in the character image.

5 Conclusions

A Gabor filter family is very similar to a wavelet family, and the most difference between them is the mode of decomposing the 2-D frequency domain: the former is circle mode, but the latter is square mode. And thus Gabor filters have the orientation characteristics. We have designed three Gabor filter families and discussed the differences in them, and we commend the third one to users. Using Gabor filter family, users can extract out many features in the image, such as feature F_1, F_2, F_3 and F_4.

The three experiments above demonstrate that these features extracted using the Gabor filter family well represent the contents in the image. And these features can be used in our practical applications. The parameters of the Gabor filter family are preset and are not necessarily optimal for a particular task.

References

[1] John G. Daugman, Complete Discrete 2-D Gabor Transforms by Neural Networks for Image Analysis and Compression, *IEEE Trans. on Acoustics, Speech and Signal Processing, 36*(7), 1988, 1169-1179.

[2] K.R. Namuduri, R. Mehrotra, and N. Ranganathan, Edge detection models based on Gabor filters, *IEEE 11th IAPR International Conf. On Speech and Signal Analysis Proceedings*, 1992, 729-732.

[3] Thomas P. Weldon, William E. Higgins and Dennis F. Dunn, Efficient Gabor filter design for texture segmentation, *Pattern Recogniton, 29*(12), 1996, 2005-2015.

[4] Chien-Chang Chen and Daniel C.Chen, Multi-resolution Gabor filter in texture analysis, *Pattern Recognition Letters, 17*(10), 1996, 1069-1076.

[5] Jianwei Yang, Lifeng Liu, Tianzi Jiang and Yong Fan, A modified Gabor filter design method for finger-

print image enhancement, *Pattern Recognition Letters*, *24*(12), 2003, 1805-1817.

[6] Jouko Lampinen and Erkki Oja, Distortion Tolerant Pattern Recognition Based on Self-Organizing Feature Extraction, *IEEE Trans. on Neural Networks, 6*(3), 1995, 539-547.

[7] V. Tavsanoglu and E. Saatci, Feature Extraction for Character Recognition Using Gabor-type Filters Implemented by Cellular Neural Networks, *6th Proc. IEEE International Workshop on CNN and Their Application*, 2000, 63-68.

[8] Su Yih-Ming, and Wang Jhing-Fa, A novel stroke extraction method for Chinese characters using Gabor filters, *Pattern Recognition, 36*(3), 2003, 635-647.

[9] Chih-Jen Lee, Sheng-De Wang, and Kuo-Ping Wu, Fingerprint Recognition Using Principal Gabor Basis Function, *Proc. of International Symposium on Intelligent Multimedia, Video and Speech Proceeding*, 2001, 393-396.

[10] Chengjun Liu and Harry Wechsler, Independent Component Analysis of Gabor Features for Face Recognition, *IEEE Trans. on Neural Networks, 14*(4), 2003, 919-928.

[11] A. Bodnarova, M. Bennamoun and S. Latham, Optimal Gabor filters for textile flaw detection, *Pattern Recognition, 35*(12), 2002, 2973-2991.