ELECTRONICS AND ELECTRICAL ENGINEERING
ISSN 1392 – 1215 *2010. No. 1(97)*
ELEKTRONIKA IR ELEKTROTECHNIKA

SYSTEM ENGINEERING, COMPUTER TECHNOLOGY
T 120
SISTEMŲ INŽINERIJA, KOMPIUTERINĖS TECHNOLOGIJOS

Accelerated Calculation of Gabor Features in Spatial Domain

J. Kranauskas
Department of Computer Science II, Faculty of Mathematics and Informatics, Vilnius University, Naugarduko str. 24, LT-03225 Vilnius, Lithuania, e-mail: justas@atrodo.com

Introduction

Gabor filters have been widely used in constructing various Gabor features for different computer vision tasks: competitive texture classification, segmentation and synthesis, fast and accurate object detection and tracking [1, 2], one of most precise biometrics iris recognition [3], and especially face recognition [4–8]. Gabor features are proved to perform very well because of their properties like rotation, scale, translation and uniform lighting semi-invariance [9]. On the other hand, computational complexity still limits their application in practice. We will focus on most widely used Gabor features - convolution with multi-resolution structure of Gabor filters of several frequencies and orientations.

A straight forward implementation of Gabor features extraction would be an image convolution with Gabor filters in spatial domain. It can be improved by an order of magnitude using the separability property of 2D filters [10] or symmetry / anti-symmetry / wavelet characteristics for special cases of Gabor filters orientations and frequencies [11]. Several schemas of calculating Gabor features more effectively by approximations were presented: effective area of filters and Laplacian pyramid [12], recursive Gabor [13], decomposition into Gaussians [14]. However, the most effective Gabor features extraction at every location in the image is done by using Fast Fourier Transform (FFT) for image convolution with Gabor filters in frequency domain. Several works in pattern recognition [15,16] use Gabor features that are calculated at some regular grid but not every pixel of the image. Motivated by that we will explore how a structure of regular grid and generalized separability of Gabor filter can be exploited to speed up the calculation of Gabor features almost to the speed of FFT without loss of precision. The main contributions of this article are:

1. Exploitation of the structure of regular grid, symmetry and generalized separability of Gabor filter to accelerate Gabor features calculation in spatial domain.
2. Calculation of whole Gabor feature at once.
3. Detailed comparison to direct convolution, proposed convolution, and convolution in frequency domain.

Gabor Filter

Following [17], we will assume complex-valued 2D Gabor filter as product of isotropic Gaussian and complex exponential plane wave:

$$G(x, y, \theta, f) = e^{-\frac{x_\theta^2 + y_\theta^2}{2\sigma^2}} \cos(2\pi f x_\theta + \varphi), \quad (1)$$

where $x_\theta = x\cos\theta + y\sin\theta$, $y_\theta = -x\sin\theta + y\cos\theta$ $\theta \in [0 \ldots \pi)$ – filter orientation; f – filter frequency; σ – a standard deviation of Gaussian function; $\varphi \in \{0, \pi/2\}$ corresponds to real and imaginary parts of Gabor filter.

We will show the Gabor filter of any orientation θ and any frequency f becomes linearly-multi-separable (formal definition will be presented later) and we will show how this and the filter symmetry properties can be exploited for accelerated calculation of Gabor features.

Effective Filter Envelope

Effective filter envelope corresponds to the filter are with significant coefficients [12]. Filter coefficients outside that area can be discarded depending on what accuracy and speed ratio is needed. Although, speed of convolution in frequency domain is not affected by smaller filter size[1], it can significantly reduce computational complexity, when filtering is performed in spatial domain, and memory consumptions for storage of filters. Effective filter envelope of Gabor filter can be calculated directly from standard deviation of approximately normally distributed data. For all further experiments doubled standard deviation will be used as the radius of effective filter envelope retaining approximately 95% energy of the filter.

Convolution

Direct convolution of a linear *M*x*M* (where $M = 2m + 1$) 2D filter C and image I in spatial domain is defined as

[1] Here we neglect the fact that convolution by FFT without a signal wraparound requires complementing of the longer signal by half of the shorter signal with zeros.

$$H(r,c)=\sum_{i=-m}^{m}\sum_{j=-m}^{m}C(i,j)I(r-i,c-j), \qquad (2)$$

which requires $O(M^2)$ (operations) calculations to calculate convolution at one point of the image and $O(M^2N^2)$ (operations) calculations for convolution with the whole NxN image. Linear 2D filter is said to be separable if it can be decomposed as a product of two one-dimensional signals filters. Convolution of whole image with separable 2D filter can be speeded up by convolving each row of the image with the horizontal projection of filter, resulting in the intermediate image. Then, convolving each column of the intermediate image with the vertical projection of filter. The resulting image is identical to direct convolution, no matter which step (horizontal or vertical) is performed first, and requires $O(MN^2)$ calculations. We will generalize notion of separable 2D filter that will be applicable for our complex valued Gabor filter.

Definition 1. It is said that complex-valued 2D filter $C(x, y)$ is *linearly multi-separable* if for some finite K_1 and K_2

$$C(x,y)=\sum_{k=1}^{K_1}a_k^1(x)b_k^1(y)+\sqrt{-1}\sum_{k=1}^{K_2}a_k^2(x)b_k^2(y), \qquad (3)$$

where $a_k^j(x)$ and $b_k^j(y)$ – real-valued functions. Sum K_1+K_2 is referred as *order* of multi-separability.

Note that NxN image filtration with linearly multi-separable filter would require $2(K_1+K_2)MN^2+(K_1+K_2-2)N^2$ arithmetic operations. We will show below that the Gabor filter (1) is multi-separable of order 4. The real part of the filter (1) can be decomposed in

$$\begin{aligned}
&\Re G(x,y,\theta,f)=\\
&=e^{-\frac{(x\cos\theta+y\sin\theta)^2+(-x\sin\theta+y\cos\theta)^2}{2\sigma^2}}\times\\
&\times\cos(2\pi f(x\cos\theta+y\sin\theta))=\\
&=e^{-\frac{x^2+y^2}{2\sigma^2}}\cos(2\pi f(x\cos\theta+y\sin\theta))=\\
&=e^{-\frac{x^2+y^2}{2\sigma^2}}\cos(2\pi fx\cos\theta)\cos(2\pi fy\sin\theta)-\\
&-e^{-\frac{x^2+y^2}{2\sigma^2}}\sin(2\pi fx\cos\theta)\sin(2\pi fy\sin\theta)=\\
&=G_{h1}(x,\theta,f)G_{v1}(y,\theta,f)-\\
&-G_{h2}(x,\theta,f)G_{v2}(y,\theta,f).
\end{aligned} \qquad (4)$$

where

$$\begin{cases}
G_{h1}(x,\theta,f)=e^{-\frac{x^2}{2\sigma^2}}\cos(2\pi fx\cos\theta),\\
G_{h2}(x,\theta,f)=e^{-\frac{x^2}{2\sigma^2}}\sin(2\pi fx\cos\theta),\\
G_{v1}(y,\theta,f)=e^{-\frac{y^2}{2\sigma^2}}\cos(2\pi fy\sin\theta),\\
G_{v2}(y,\theta,f)=e^{-\frac{y^2}{2\sigma^2}}\sin(2\pi fy\sin\theta).
\end{cases} \qquad (5)$$

Similarly, the imaginary part of Gabor filter can be decomposed in

$$\begin{aligned}
&G(x,y,\theta,f)=G_{h2}(x,\theta,f)G_{v1}(y,\theta,f)+\\
&+G_{h1}(x,\theta,f)G_{v2}(y,\theta,f).
\end{aligned} \qquad (6)$$

Combining (3) and (5) we have that complex Gabor filter is linearly multi-separable of order 4. Using additional benefits of symmetry of (3) and (5) expressions we can reduce the filtration complexity $8MN^2+6N^2$ to $6MN^2+2N^2$. The last significantly improves complexity of direct convolution consisting of $4M^2N^2$ arithmetic operations.

Convolution H in frequency domain is done by converting image I to frequency domain with FFT, multiplying by a converted to frequency domain filter C and converting back to the spatial domain with IFFT:

$$H = IFFT(FFT(I)\cdot FFT(F)). \qquad (7)$$

This approach gives a periodic version of convolution. To obtain non-periodic convolution, the approach of filtering in frequency domain requires a modification involving an additional complexity. For the simplicity of analysis we will restrict ourselves on periodic version of convolution. Computational complexity of FFT, as well as IFFT, is $O(N^2\log N)$, however the lowest bound of the exact count of arithmetic operations of 1D FFT (split-radix FFT algorithm [18]) is $4N\log_2 N-6N+8$ real additions and multiplications[2] and applies only for N a power of two greater than 1. Since complexity of convolution in frequency domain does not depend on filter size (if filter is smaller than the whole image) and complex Gabor filter can be computed directly in frequency domain by

$$\Gamma(u,v,\theta,f)=2\pi\sigma^2e^{-(2\pi\sigma)^2\left((u-f\cos\theta)^2+(v-f\sin\theta)^2\right)}. \qquad (8)$$

Arithmetic complexity of convolution with complex Gabor filter is $4N^2\log_2 N-4N^2+8N$ (one IFFT of 2D signal plus one complex multiplication in frequency domain, if image and filter are already in frequency domain[3]).

Multi-resolution and Multi-orientation

The most attractive property of Gabor feature – orientation and scale semi-invariance – is achieved by using Gabor filters of many different orientations and scales which describe local structure of the image. In [12], Laplacian pyramid of images for faster calculation of multi-resolution Gabor feature is suggested. However, speed improvement comes with several drawbacks, and once again works with only very special case of Gabor features:

1. Pyramid of images can be effectively constructed only for integer scaling factors, downscaling by real factors creates aliasing effect and should be avoided.

2. Responses of Gabor filters are approximated and, additionally, special care should be taken to upscale the

[2] Recently it was improved to $\approx\frac{34}{9}N\log_2 N$ [19].

[3] The complexity of image conversion to frequency domain is not added here because further we analyze a Gabor feature (which is composed of several Gabor filters) calculation time.

responses of low frequency filters back to the higher resolution.

We will focus on calculating exact values of Gabor filters of equally distributed directions (covering the $[0\ldots\pi]$ range with a constant step) which are not limited to particular scales that are convenient for constructing Laplacian pyramid of images. Authors of [11] try to calculate convolution with several Gabor filters at once, but is tuned to orientations with $\theta \in \{0, \pi/4, 2\pi/4, 3\pi/4\}$ and 3 scales (starting from 3×3 discrete filter and scaling it by a factor of 2). Most often each Gabor filter is dealt with independently in other works. One of the main contributions of the proposed method is calculating the whole Gabor feature at once in spatial domain, including convolution with Gabor filters of all equally distributed directions and any number of scales.

Regular Grid

A regular grid is used in some pattern recognition tasks like [15, 16]. Proposed optimizations are most effective if Gabor features are calculated at adjacent points which are closer than half of the largest Gabor filter. Several examples of regular grids are shown in Fig. 1 (they are linearly separable into horizontal and vertical parts).

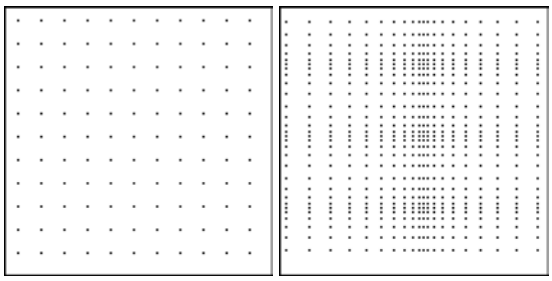

Fig. 1. Examples of regular grids in the 128x128 image

Implementation

A classical example is a vector of responses got from image convolution with Gabor filters of 8 orientations and 5 frequencies homogeneously distributed in a frequency band [5]. Number of frequencies is not limited by proposed implementation, however number of orientations should be even to use full ensemble of optimizations. On the other hand, this limitation is not exceptional because almost every application of Gabor features in the literature uses even number of orientations. In previous section we showed that every Gabor filter with isotropic Gaussian part is multi-separable of order 4 and this can be exploited to speed up the convolution in spatial domain. Further we will show how symmetry and anti-symmetry of Gabor filter as well as Gabor feature can be used for speeding up the convolution at any location in the image up to four times.

Symmetry (Anti-symmetry) of Gabor Filter

Real and imaginary parts of Gabor filters are symmetric and anti-symmetric – they have the same modulus at locations that components have equal absolute values (Fig. 2).

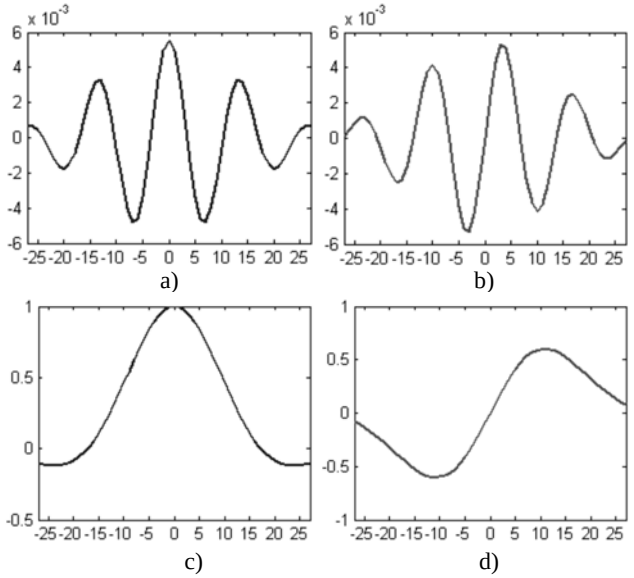

Fig. 2. G_{h1} (a), G_{h2} (b), G_{v1} (c) and G_{v2} (d) parts of Gabor filter with orientation $\theta = \pi/16$

Direct convolution of 1D signal I and filter $F = (f_0, \ldots, f_M)$, where $f_i = f_{M-i}$, at location x (which requires $M + 1$ multiplications and M additions)

$$H(x) = f_0 I(x-m) + f_1 I(x-m+1) + \\ + \ldots + f_{M-1} I(x+m-1) + f_M I(x+m). \quad (9)$$

can be replaced by symmetric (or anti-symmetric if sums of signal values will be replaced by differences) version (which requires $m + 1$ multiplications and M additions):

$$H_s(x) = f_{m+1} I(x) + f_m (I(x-1) + I(x+1)) + \\ + \ldots + f_0 (I(x-m) + I(x+m)). \quad (10)$$

Although, in [12] authors state that this will not lead to any improvement on modern computers because multiplication is not an expensive operation, symmetric convolution requires 25% less arithmetic operations which can reduce convolution time with one filter by one fourth. Additionally, when symmetric convolution is used to calculate Gabor feature, each Gabor filter can use the same sums (or differences) of signal and they can be precalculated only once for the largest filter. Another 25% of arithmetic operations can be saved for all but one largest filter.

Symmetry (Anti-symmetry) of Gabor Feature

One more symmetry (anti-symmetry) exists in Gabor feature between filters of same scale but different orientations. If D orientations are used in Gabor feature, full convolution must be performed only with filters of $\frac{D}{2}+1$ orientations that fall in the range $\left[0, \ldots, \frac{\pi}{4}\right]$. The remaining filters are symmetric (anti-symmetric) and their responses can be calculated by reusing previously calculated filters responses as shown below for G_{h1} filter part:

$$G_{h1}(x, \pi-\theta, f_0) = $$
$$= e^{-\frac{x^2}{2\sigma^2}} \cos(2\pi f_0 x \cos(\pi-\theta)) =$$
$$= e^{-\frac{x^2}{2\sigma^2}} \cos(-2\pi f_0 x \cos\theta) =$$
$$= e^{-\frac{x^2}{2\sigma^2}} \cos(2\pi f_0 x \cos\theta) = G_{h1}(x, \theta, f_0). \quad (11)$$

Similarly,

$$\begin{cases} G_{h2}(x, \pi-\theta, f_0) = -G_{h2}(x, \theta, f_0), \\ G_{v1}(y, \pi-\theta, f_0) = G_{v1}(y, \theta, f_0), \\ G_{v2}(y, \pi-\theta, f_0) = G_{v2}(y, \theta, f_0). \end{cases} \quad (12)$$

Responses of corresponding Gabor filters can be calculated by:

$$G(x, y, \pi-\theta, f_0, 0) = $$
$$= G_{h1}(x, \theta, f_0) G_{v1}(y, \theta, f_0) +$$
$$+ G_{h2}(x, \theta, f_0) G_{v2}(y, \theta, f_0), \quad (13)$$

$$G(x, y, \pi-\theta, f_0, \tfrac{\pi}{2}) = $$
$$= -G_{h2}(x, \theta, f_0) G_{v1}(y, \theta, f_0) +$$
$$+ G_{h1}(x, \theta, f_0) G_{v2}(y, \theta, f_0). \quad (14)$$

Using symmetry along orientations $\frac{D}{2}-1$ of orientations won't be recalculated and will save almost 50% arithmetic operations if more than two orientations will be used.

Filtering at Regular Grid

Image convolution with a linear separable 2D filter can be optimized by an order of magnitude exploiting the filter multi-separability property. Actually, convolution of the whole image is the same as convolution at the dense regular grid which has a distance of one pixel between the adjacent grid positions. Same optimizations are possible if the distance between adjacent grid positions is greater than one pixel (but not greater than the length of the filter). Often there is no need to have Gabor response at every location in the image but calculating direct convolution is time consuming and FFT must be used.

Filtering Near Image Boundary

Several practices how filtering near image boundaries could be dealt with, when part of the filter slips outside the image, come from image processing:

1. Extend the image with a constant (possibly zero) intensity value.
2. Extend the image periodically or by mirroring it at the boundaries.
3. Normalize the response of convolution by sum of values from the filter part which does not slip outside the image.

Discrete Gabor filters are constructed to have a DC free property, i. e. sum of filter coefficients equals to zero. When part of the filter slips outside the image, Gabor filter loses the DC free property and its response can change unacceptably. Filter response normalization is necessary and can be done by subtracting the DC value. We calculate a DC value by applying integral image technique which enables rapid calculation of sums of values in any rectangular region in a constant time. During our experiments we noticed that such approach can significantly improve Gabor filter response stability near image boundary even when up to 45% of the filter is outside the image and is compensated by DC value (Fig. 4).

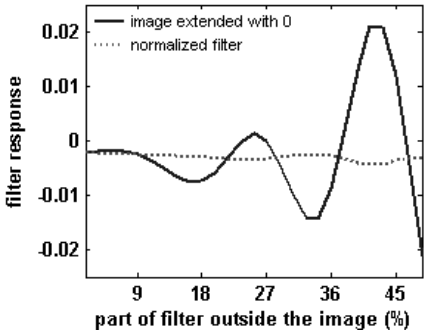

Fig. 3. One specific Gabor response when part of the image under the filter is not available (with and without filter response normalization). This simulates filter response stability near image boundary

Evaluation

First of all, arithmetic complexity of the proposed method is compared to direct and FFT based methods by calculating number of arithmetic operations required to perform Gabor features extraction at regular grids of different sizes. Regular grids were chosen in the following order: each point, every second point, every third point, …, one center point in the image. The proposed method should always be faster than direct calculation of Gabor features and should be faster than convolution in frequency domain when Gabor features are needed only on every second point of the image (Fig. 4).

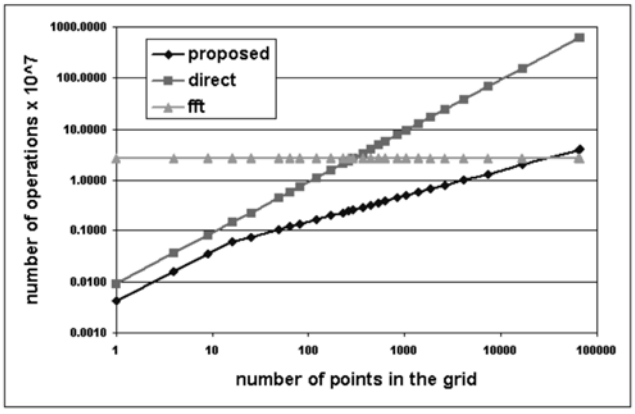

Fig. 4. Number of arithmetic operations required for the evaluated methods to calculate Gabor features (9 scales, 16 orientations) at regular grids of different sizes on 256x256 image

Complexity of convolution in frequency domain was calculated according to the split-radix FFT method that is applicable only to the signals of power-of-two length. In

practice its complexity heavily depends on effectiveness of implementation. For further experiments one of the most efficient publicly available FFT software – FFTW library [20] will be used. Results of practical experiment with the same Gabor features, image size and regular girds that were used in theoretical evaluation can be seen in Fig. 5.
Different time for the same number of points for the direct Gabor features calculations appear from the regular grids where the same points are situated further or closer to image boundary. Gabor filters from the features that are calculated closer to the image boundary slip outside the image and are calculated faster because parts of them are not used in convolution. However this does not affect the speed of the proposed method because the problem of image boundary is solved at the precalculation of signal sums (and differences) step. One more difference from theoretical evaluation of complexity is the form of the proposed method curve. This can be explained by the fact that arithmetic complexity was calculated without taking into account (actually, taking the worst case) how Gabor filters overlap in the regular grid, i. e. how close adjacent points in the regular grid are.

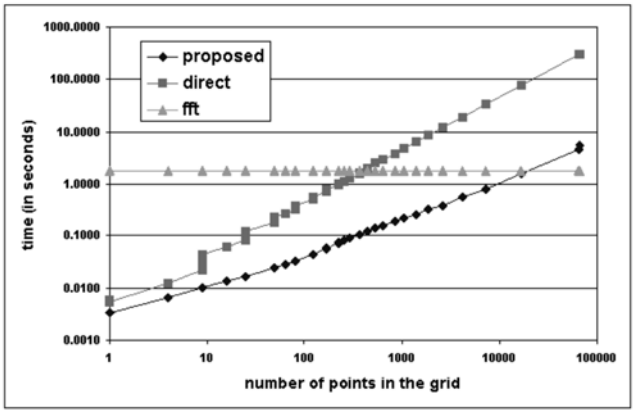

Fig. 5. Time (in seconds) required for the evaluated methods to calculate Gabor features (9 scales, 16 orientations) at regular grids of different sizes on 256x256 image

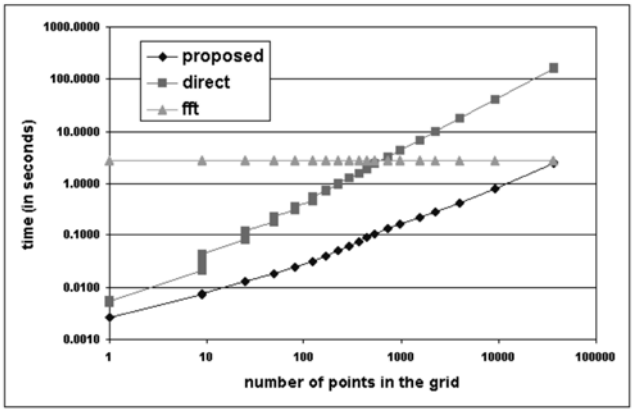

Fig. 6. Time (in seconds) required for the evaluated methods to calculate Gabor features (9 scales, 16 orientations) at regular grids of different sizes on 191x191 image

Images with dimensions of power-of-two are very convenient for the FFT. To show the efficiency of the proposed method the same experiment was performed with image of 191x191 pixels, results can be seen in Fig. 6. Now the proposed method outperforms convolution in frequency domain by 10% even at calculating Gabor features at each point of the image (though the difference is only marginal in logarithmic scale).

Similar experiments were performed with different number of scales (3, 5, 9) and orientations (4, 8, 16, 32) in Gabor features and different sizes of images (128x128, 191x191, 256x256). The results are almost identical to those that were presented above.

Conclusion

The regular grid together with Gabor filters symmetry (anti-symmetry) and Gabor features symmetry (anti-symmetry) along directions were successfully used to improve Gabor features calculation time. The problem of filtering near image boundary was also addressed and efficient solution was proposed. After the detailed comparison of the proposed method with direct and FFT based calculation of Gabor features the following conclusions can be drawn from the evaluation results:

1. Proposed method is always faster than the direct convolution.
2. Proposed method is faster than convolution in frequency domain if Gabor features are required at every second (sometimes every third) point of the image and image dimensions are convenient for FFT.
3. Proposed method is always faster than convolution in frequency domain if image dimensions are not convenient for FFT.

Several approaches of optimizations were left out of the scope of this article and will be analyzed in our future works. Firstly, it is exploiting Gabor wavelet property, i. e. different Gabor scales are generated from one mother wavelet. Secondly, modern processors are able to do several arithmetic operations in parallel and this can be used for parallel version of the proposed method. And lastly, various approximations of Gabor filters could be used to speed up features extraction with small errors in Gabor responses. If the errors are too small to influence the quality of further steps in the algorithms, approximations should be definitely used.

References

1. **Mojaev A., Zell A.** Real–Time Object and Face Tracking with Gabor Wavelets // Proceedings of the 11th IEEE ICAR. – Portugal, Coimbra. – 2003. – P. 1178–1183.
2. **Jain A. K., Ratha N. K.** Object Detection Using Gabor Filters // Pattern Recognition. – 1997. – Vol. 13, No. 2. – P. 295–309.
3. **Daugman J.** Demodulation by Complex–Valued Wavelets for Stochastic Pattern Recognition // International Journal of Wavelets, Multiresolution and Information Processing. – 2003. – Vol. 1, No. 1. – P. 1–17.
4. **Zou J., Ji Q., Nagy G.** A Comparative Study of Local Matching Approach for Face Recognition // IEEE TPAMI. – 2007. – Vol. 16, No. 10. – P. 2617–2628.
5. **Wiskott L., Fellous J. M., Kruger N., Malsburg C.v.d.** Face recognition by elastic bunch graph matching // IEEE TPAMI. – 1997. – Vol. 19, No. 7. – P. 775–779.
6. **Zhang W., Shan S., Gao W., Chen X., Zhang H.** Local Gabor binary pattern histogram sequence (LGBPHS): A

novel non–statistical model for face representation and recognition // in Proceedings ICCV. – 2005. – P. 786–791.
7. **Su Y., Shan S., Chen X., Gao W.** Patch–Based Gabor Fisher Classifier for Face Recognition // 18th International Conference on Pattern Recognition. – 2006. – P. 528–531.
8. **Tan X., Triggs B.** Fusing Gabor and LBP feature sets for kernel–based face recognition // In Analysis and Modelling of Faces and Gestures. – 2007. – P. 235–249.
9. **Kyrki. V., Kamarainen J.–K.** Simple Gabor Feature Space for Invariant Object Recognition // Research Report 83, Lappeenranta University of Technology, Department of Information Technology. – 2003.
10. **Areekul V., Watchareeruetai U., Suppasriwasuseth K., Tantaratana S.** Separable Gabor Filter Realization for Fast Fingerprint Enhancement // IEEE International Conference on Image Processing. – 2005. – Vol. 3. – P. 253–256.
11. **Ranganathan N., Mehrotra R., Namuduri K. R.** An Architecture to Implement Multiresolution // Proceedings of the International Conference on Acoustics, Speech, and Signal Processing. – 1991. – Vol. 2. – P. 1157–1160.
12. **Ilonen J., Kamarainen J. K., Kälviäinen H.** Efficient Computation of Gabor Features // Research Report 100, Lappeenranta University of Technology, Department of Information Technology. – 2005.
13. **Young I. T., Vliet L. J., Ginkel M.** Recursive Gabor Filtering // IEEE Transactions on Signal Processing. – 2002. – Vol. 50, No. 11. – P. 2798–2805.
14. **Bernardino A., Santos–Victor J.** A Real–Time Gabor Primal Sketch for Visual Attention // 2nd Iberian Conference on Pattern Recognition and Image Analysis. – Estoril, Portugal, 2005.
15. **Walter J., Arnrich B., Scherring C.** Learning Fine Positioning of Robot Manipulator Based on Gabor Wavelets // Proceedings of the International Joint Conference on Neural Networks. – 2000. – Vol. 5. – P. 137–142.
16. **Westphal G., Malsburg C.v.d., Wurtz R.P.** Feature–driven Emergence of Model Graphs for Object Recognition and Categorization // Applied Pattern Recognition. – 2008. – P. 155–199.
17. **Wiskott L., Fellous J.M., Krüger N., Malsburg C.v.d.** Face Recognition by Elastic Bunch Graph Matching // IEEE TPAMI. – 1997. – Vol. 19, No. 7. – P. 775–779.
18. **Yavne R.** An economical method for calculating the discrete Fourier transform // Proceedings AFIPS Fall Joint Comput. Conf. – 1968. – Vol. 33. – P. 115–125.
19. **Johnson S. G., Frigo M.** A Modified Split–Radix FFT With Fewer Arithmetic Operations // IEEE Transactions on Signal Processing. – 2007. – Vol. 55, No. 1. – P. 111–119.
20. **Frigo M., Johnson S. G.** The Design and Implementation of FFTW3 // Proceedings of the IEEE. – 2005. – Vol. 93, No. 2. – P. 216–231.

Received 2009 09 10

J. Kranauskas. Accelerated Calculation of Gabor Features in Spatial Domain // Electronics and Electrical Engineering. – Kaunas: Technologija, 2010. – No. 1(97). – P. 39–44.

A lot of computer vision tasks are tried to solve by using Gabor features that proved to be very effective feature descriptors. One of the main drawbacks of using Gabor features in real-time practical applications is computational heaviness. Convolution with Gabor filters in frequency domain generates responses of Gabor features at each position in the image and is time consuming operation when Gabor feature consists of tens of Gabor filters. In this article, we introduce the concept of calculating Gabor features at regular grid in the image and show how this together with a generalized separability, symmetry and anti-symmetry of Gabor filter can be exploited to calculate Gabor features in spatial domain even faster than in frequency domain. Ill. 6, bibl.20 (in English; abstracts in English, Russian and Lithuanian).

Ю. Кранаускас. Ускоренный расчет признаков Габора в пространственной области // Электроника и электротехника. – Каунас: Технология, 2010. – № 1(97). – С. 39–44.

Признаки Габора часто используют для решения задач компьютерного зрения, т. к. они оказались очень эффективными дескрипторами признаков. Сложность вычисления – один из главных недостатков использования признаков Габора для решения практических задач реального времени. Свёртка фильтров Габора в частотной области генерирует ответ признаков Габора в каждой позиции изображения и требует много расчётов, когда признаки Габора состоят из десятков фильтров Габора. В этой статье представляется новый способ расчёта в изображении регулярно расположенных признаков Габора, и показано, как это можно сделать в пространственной области эффективнее, чем в частотной области, если использовать обобщённую отделимость, симметрию и антисимметрию фильтров Габора. Ил. 6, библ. 20 (на английском языке; рефераты на английском, русском и литовском яз.).

J. Kranauskas. Ypač spartus Gaboro požymių skaičiavimas erdvinėje srityje // Elektronika ir elektrotechnika. – Kaunas: Technologija, 2010. – Nr. 1(97). – P. 39–44.

Gaboro požymiai yra itin efektyvūs požymių deskriptoriai, todėl dažnai naudojami kompiuterinės regos algoritmuose. Taikyti Gaboro požymius praktiniams uždaviniams spręsti yra brangu, nes dėl labai sudėtingų skaičiavimų negalima kurti realiu laiku veikiančių sistemų. Sparčiausiai Gaboro požymiai apskaičiuojami dažnių srityje. Taigi Gaboro požymių atsakai gaunami kiekviename paveiksliuko taške ir daugiausia dėl to tai vis dar yra skaičiavimų reikalaujantis sprendimas, kai Gaboro požymius sudaro dešimtys Gaboro filtrų. Šiame straipsnyje pristatomas naujas būdas skaičiuoti reguliariai paveiksliuke išsidėsčiusius Gaboro požymius bei parodoma, kaip, naudojant Gaboro filtrų apibendrintąjį atskiriamumą, simetriją ir antisimetriją, ypač efektyviai tai galima atlikti erdvinėje srityje. Il. 6, bibl. 20 (anglų kalba; santraukos anglų, rusų ir lietuvių k.).

DOI: 10.5755/j02.eie.9942