MVA2011 IAPR Conference on Machine Vision Applications, June 13-15, 2011, Nara, JAPAN

Fast Polar Cosine Transform for Image Description

Zhuo YANG and Sei-ichiro KAMATA
Graduate School of Information,Production and System, Waseda University
joel@ruri.waseda.jp and kam@waseda.jp

Abstract

Polar Cosine Transform (PCT) is one of the Polar Harmonic Transforms that those kernels are basic waves and harmonic in nature. They are proposed to represent invariant patterns for two dimensional image description and are demonstrated to show superiorities comparing with other methods on extracting rotation invariant patterns for images. However in order to increase the computation speed, fast algorithm for PCT is proposed for real world applications like limited computing environments, large image databases and real-time systems. Based on our previous work, this paper novelly employs relative prime number theory to develop Fast Polar Cosine Transform (FPCT). The proposed FPCT is averagely over 11 $\sim$ 12.5 *times faster than PCT that significantly boost computation process. The experimental results are given to illustrate the effectiveness of the proposed method.*

1 Introduction

Rotation invariant pattern representation is one of the essential challenges in image retrieval and pattern recognition arises from the fact that in many real world applications, images should be considered to be the same even if they are rotated. Polar Cosine Transform (PCT) as one of Polar Harmonic Transforms (PHTs) is proposed to represent two dimensional images and demonstrated to show superiorities comparing with other methods [1]. With the orthogonal property, PC-T can transform the image function to a set of mutually independent patterns with minimum redundancy and maximal discriminant information. Unfortunately, kernel generation of these transforms involves many trigonometric functions to compute angular and radial parts that no fast method has been reported to best of our knowledge. The high computational complexity is the constraint for real world applications such as realtime systems, limited computing environments and large image databases. Therefore, reduction of the computational complexity for PCT is very significant.

Previous work proposes fast PHTs [2]. Fast and compact methods to compute the kernel coefficients of PHTs are proposed by using mathematical properties of trigonometric functions. The kernel function of PHTs has symmetry properties with respect to the x axis, y axis, $y = x$ line, $y = -x$ line and origin that can be used for fast computation. The computational complexity of PHTs can be reduced by calculating half of the first quadrant. That is only one eighth of the direct transform.

In this paper,an even more efficient kernel computation method based on relative prime number theory is proposed to compute PCT after analyzing the point distribution on two dimensional discrete space. Much more symmetric points are involved to compute simultaneously in order to significantly accelerate the transform speed. This is named as Fast Polar Cosine Transform (FPCT). Due to the paper length limitation, PC-T is mainly discussed in this paper. The method we discussed can also be used in other PHTs.

The organization of this paper is as follows. The basic theories of PCT and fast algorithms including mathematics descriptions are provided in Section 2. The proposed method is presented in Section 3 after given the mathematical properties of relative prime number theory. In Section 4, the performance of PC-T and FPCT are compared against different images. The experimental results illustrate the effectiveness of proposed method. Finally, concludes this study.

2 Background

This section introduces the background of PCT and fast algorithm .

2.1 Polar Cosine Transform

Given a 2D image function $f(x, y)$, it can be transformed from cartesian coordinate to polar coordinate $f(r, \theta)$,where r and θ denote radius and azimuth respectively. The following formulae transform from cartesian coordinate to polar coordinate,

$$r = \sqrt{x^2 + y^2}, \tag{1}$$

and

$$\theta = \arctan(\frac{y}{x}). \tag{2}$$

It is defined on the unit circle that $r \leq 1$, and can be expanded with respect to the basis functions $H^C_{nl}(r, \theta)$ as

$$\boldsymbol{f}(r, \theta) = \sum_{n=0}^{\infty} \sum_{l=-\infty}^{\infty} \boldsymbol{M}^C_{nl} H^C_{nl}(r, \theta), \tag{3}$$

where the coefficient is

$$\boldsymbol{M}^C_{nl} = \Omega_n \int_0^{2\pi} \int_0^1 \boldsymbol{f}(r, \theta) H^{C*}_{nl}(r, \theta) r dr d\theta. \tag{4}$$

The basis function is given by

$$H^C_{nl}(r, \theta) = \boldsymbol{R}^C_n(r) e^{il\theta}, \tag{5}$$

where

$$\boldsymbol{R}^C_n(r) = cos(\pi n r^2), \tag{6}$$

and

$$\Omega_n = \begin{cases} \frac{1}{\pi} & \text{if } n = 0 \\ \frac{2}{\pi} & \text{if } n \neq 0 \end{cases} \tag{7}$$

and satisfying orthogonality condition

$$\int_0^1 \boldsymbol{R}^C_n(r) \boldsymbol{R}^{C*}_{n'}(r) r dr = \frac{1}{2} \delta_{nn'}, \tag{8}$$

$$G_l(x,y) = \begin{cases} (f(x,y)+f(y,x)+f(-y,x)+f(-x,y) \\ +f(-x,-y)+f(-y,-x)+f(y,-x)+f(x,-y))cos(l\theta) & \text{if } mod(l,4)=0 \\ (f(x,y)-f(-x,y)-f(-x,-y)+f(x,-y))cos(l\theta) \\ +(f(y,x)-f(-y,x)-f(-y,-x)+f(y,-x))sin(l\theta) & \text{if } mod(l,4)=1 \\ (f(x,y)-f(y,x)-f(-y,x)+f(-x,y) \\ +f(-x,-y)-f(-y,-x)-f(y,-x)+f(x,-y))cos(l\theta) & \text{if } mod(l,4)=2 \\ (f(x,y)-f(-x,y)-f(-x,-y)+f(x,-y))cos(l\theta) \\ -(f(y,x)-f(-y,x)-f(-y,-x)+f(y,-x))sin(l\theta) & \text{if } mod(l,4)=3 \end{cases} \quad (13)$$

$$H_l(x,y) = \begin{cases} (f(x,y)-f(y,x)+f(-y,x)-f(-x,y) \\ +f(-x,-y)-f(-y,-x)+f(y,-x)-f(x,-y))sin(l\theta) & \text{if } mod(l,4)=0 \\ (f(x,y)+f(-x,y)-f(-x,-y)-f(x,-y))sin(l\theta) \\ +(f(y,x)+f(-y,x)-f(-y,-x)-f(y,-x))cos(l\theta) & \text{if } mod(l,4)=1 \\ (f(x,y)+f(y,x)-f(-y,x)-f(-x,y) \\ +f(-x,-y)+f(-y,-x)-f(y,-x)-f(x,-y))sin(l\theta) & \text{if } mod(l,4)=2 \\ (f(x,y)+f(-x,y)-f(-x,-y)-f(x,-y))sin(l\theta) \\ -(f(y,x)+f(-y,x)-f(-y,-x)-f(y,-x))cos(l\theta) & \text{if } mod(l,4)=3 \end{cases} \quad (14)$$

and

$$\int_0^{2\pi}\int_0^1 H_{nl}^C(r,\theta)H_{n'l'}^{C*}(r,\theta)rdrd\theta = \pi\delta_{nn'}\delta_{ll'}, \quad (9)$$

where δ_{ij} is Kronecker delta. Rewrite (11) with (12)-(14),

$$\boldsymbol{M}_{nl}^C = \Omega_n \int_0^{2\pi}\int_0^1 \boldsymbol{f}(r,\theta)\cos(\pi n r^2) (\cos(l\theta) - i\sin(l\theta))rdrd\theta \quad . \quad (10)$$

PCT is defined on unit circle. $\left|M_{nl}^C\right|$ is rotation invariant. PCT need 3 trigonometric functions to generate kernel coefficient for each point.

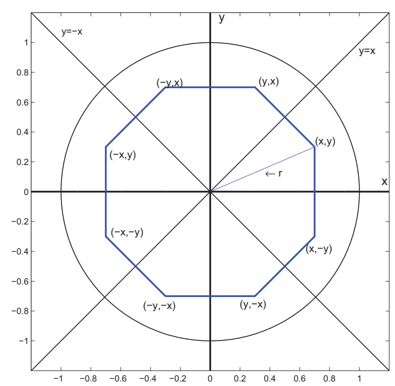


Figure 1: Symmetric points on 2D space

2.2 Fast algorithm for PCT

From Polar Cosine Transform Eq. 10, we can find that for the points on the same radius r, the different integrand part for each point is $\boldsymbol{f}(r,\theta)(\cos l\theta - i\sin l\theta)$. As Fig. 1 shown, point (x,y) is a point in first quadrant between $y=x$ and x axis, has other seven symmetric points with respect to x axis, y axis, $y=x$, $y=-x$ and origin.

As known $sin(\theta)$ and $cos(\theta)$ functions are periodic functions with period 2π. Periods for $sin(l\theta)$ and $cos(l\theta)$ are $2\pi/l$. Derived from the periodic and symmetric properties of trigonometric functions that used in fast Fourier transform [3], there are mathematical relationships for trigonometric functions respect to different l. Similar relationships also exist for cosine function and other l values. For the eight symmetric points on the same radius r, if their PCT coefficients can be calculated simultaneously, then the computation time for trigonometric function can be reduced.

Based on foregoing discussion, fast PCT is given by

$$\boldsymbol{FastM}_{nl}^C = \Omega_n \iint_D cos(\pi n(x^2+y^2)) (G_l(x,y) - iH_l(x,y))dxdy \quad , \quad (11)$$

where

$$D = \{(x,y) | 0 \le x \le 1, 0 \le y \le x, 0 \le x^2+y^2 \le 1\} \quad , \quad (12)$$

and $G_l(x,y)$ and $H_l(x,y)$ are given in Eq. 13 and 14. By using this equation, the whole PCT can be generated by using part of the basic functions.

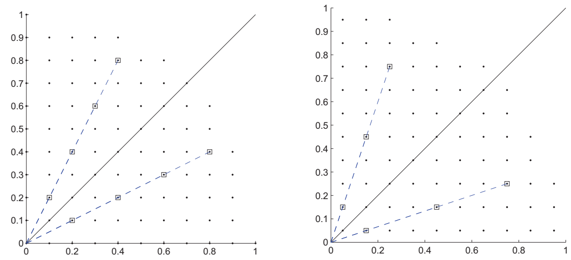
By directly computing the PCT kernel, it takes 24 trigonometric functions for symmetric eight points. By using fast PCT equation Eq. 11, computational complexity is reduced, only one eighth of the trigonometric is computed.

3 Fast Polar Cosine Transform

Foregoing proposed algorithm significantly boost the computation speed. Whether it is possible to make PCT much faster is an interesting question. Inspired by number theory [4, 5, 6], this subsection presents even faster PCT that involve much more symmetric points calculated simultaneously, and finally named as Fast Polar Cosine Transform (FPCT).

Given two integers a and b, with at least one of these being nonzero. The largest positive integer that divides both a,b is termed as the greatest common divisor of a and b.

$$gcd(a,b). \quad (15)$$



(a) Odd number size (b) Even number size

Figure 2: Odd and even number size image mapping in the first quadrant.

Here are some examples $gcd(2,6) = 2$,$gcd(3,5) = 1$ and $gcd(3,8) = 1$.

Given two integers a and b, they are said to be relative prime if their greatest common divisor is 1. They are defined [4] by

$$a \perp b, \qquad \text{if } gcd(a,b) = 1. \tag{16}$$

Conventionally 1 is relative prime to any other positive integer [5].

$$1 \perp a, \qquad \text{if } a \in N. \tag{17}$$

Given an $N \times N$ size image, there are two steps needed to transform from image conventional cartesian coordinate to PCT defined normalized unit coordinate. First, move the origin from left upper corner of image to the center. The transform equation of a point $P(X_p, Y_p)$ from original coordinate to its corresponding centered coordinate (X_c, Y_c) is given by

$$\begin{aligned} &CartesianToCenter(X_p, Y_p) \\ &= (X_p - \frac{N-1}{2}, \frac{N-1}{2} - Y_p) = (X_c, Y_c). \end{aligned} \tag{18}$$

Second, the centered coordinate is normalized to unit. The transform equation from centered coordinates to normalized is

$$CenterToUnit(X_c, Y_c) = (\frac{2X_c}{N-1}, \frac{2Y_c}{N-1}) = (x, y), \tag{19}$$

and its reverse transform equation is

$$\begin{aligned} &UnitToCenter(x, y) \\ &= (\frac{(N-1)x}{2}, \frac{(N-1)y}{2}) = (X_c, Y_c). \end{aligned} \tag{20}$$

For example in 21×21 size image, cartesian coordinates (X_p, Y_p) are $(12,9), (14,8)$ and $(16,7)$. Based on equation (39), after moving origin to center of image their coordinates (X_c, Y_c) equal to $(2,1), (4,2)$ and $(6,3)$. Based on equation (40), after normalized to unit their coordinates (x, y) are $(0.2, 0.1), (0.4, 0.2)$ and $(0.6, 0.3)$. Fig. 2 shows the first quadrant of 21×21 size image after mapping to unit circle. We define (x, y) is a relative prime point if satisfied

$$rpp(x,y) = \begin{cases} X_c \perp Y_c, & \text{if } N \text{ is odd} \\ 2X_c \perp 2Y_c, & \text{if } N \text{ is even} \end{cases}. \tag{21}$$

Given a relative prime point (x, y), for odd number size image, the points set in same angle can be represented by $\{(mx, my) | m \in N\}$, for even number size image, they can be represented by $\{((2m-1)x, (2m-1)y) | m \in N\}$. The relative prime points distributions of odd number size image and even number size image are different as shown in Fig. 2. When generating kernel of PCT, no need to generate the angular part if a point is not a relative prime point. Table 1 gives a distribution of relative prime points within a circle with different size radius.

Table 1: Distribution of Relative Prime Points in Odd Number Size Image

Radius	Relative Prime Points	Probability
1-200	9544	0.614434
201-400	28657	0.610321
401-600	47746	0.609277
601-800	66847	0.608879
801-1000	85927	0.608544

Table 2: Distribution of Relative Prime Points in Even Number Size Image

Radius	Relative Prime Points	Probability
1-199	3186	0.818392
201-399	9550	0.813043
401-599	15918	0.811977
601-799	22273	0.811491
801-999	28655	0.811412

Table 2 gives a distribution of relative primes but only for odd numbers. This is useful for even number size image. There is theoretical proof [6] to show the probability of two randomly given integers

$$p(a \perp b) = \frac{1}{\zeta(2)} = \frac{6}{\pi^2} \approx 0.607927102 \approx 61\%, \tag{22}$$

where $\zeta(z)$ refers to the Riemann zeta function. From Table 1, 2 and Eq. 22, we can find that large number of points are not relative prime points, that means their angular part is not needed to computed when generating the PCT kernel coefficients.

Based on foregoing discussion, faster PCT is given by

$$\begin{aligned} \boldsymbol{FasterM}_{nl}^{C} = \Omega_n \iint\limits_{A} \sum_{k=1}^{K} \{cos(\pi n k^2 (x^2 + y^2)) \\ (G_l(kx, ky) - iH_l(kx, ky)\} dxdy \end{aligned}, \tag{23}$$

where

$$K = \lfloor \frac{1}{\sqrt{x^2 + y^2}} \rfloor, \tag{24}$$

and

$$\begin{aligned} A = \{(x,y) | 0 \le x \le 1, 0 \le y \le x, \\ 0 \le x^2 + y^2 \le 1, rpp(x,y)\} \end{aligned}, \tag{25}$$

$\lfloor x \rfloor$ is floor function that return integral part of x. Given a point (x, y) that is a relative prime point, then by multiplying a factor k all the coordinates (kx, ky) that are in

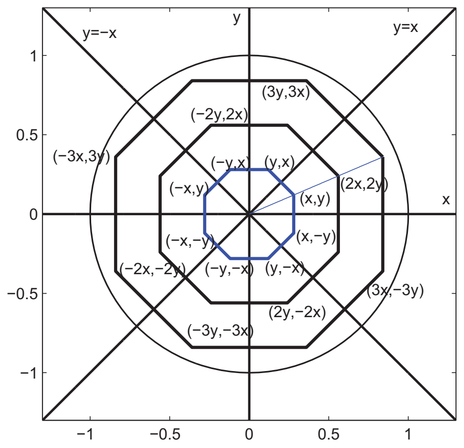


Figure 3: symmetric relative prime points.

the same angle can be obtained. Fig 3 gives an example to compute 24 points together.

By using Eq. 23 much more symmetric points are involved, and only small number of computation is needed to generate the kernel. We name Eq. 23 as FPCT.

3.1 Computation Complexity Analysis

Given a two dimensional function $f(x, y)$, inverse trigonometric function is needed to compute the polar coordinates using equation (1) and (2), and trigonometric function is needed to compute the coefficients of PCT. We use DM to denote direct transform PCT method (section 2.1), SM to denote fast PCT method that using symmetric properties [2] and RM to denote the proposed FPCT method that use relative prime points (section 3). As a summary , computation complexities in terms of number of trigonometric function and inverse trigonometric function needed to generate kernel for symmetric eight points is given in table 3. FPCT which is based on RM is the fastest.

Table 3: Computation Complexity for PCT Kernel

Functions	DM	SM	RM
Trigonometric	24	3	$1 + 2p$
Inverse Trigonometric	8	1	p

p=probability of relative prime points

4 Experimental Results

The performance of the proposed fast transforms for PC-T in computation reduction is validated through comparative experiments using various images. DM, SM and RM based transforms are all evaluated. Images with different resolutions and contents are tested to illustrate the effectiveness and feasibility of the proposed fast transforms.

Because the relative prime point distribution is different under odd and even number size images. In the experiment both odd and even number size images are used to test the effectiveness of proposed method. 20 coefficients are computed for PCT. With same computation result, DM , SM and RM based transforms have different running time. Their computation performances in terms of CPU elapsed time are given in Table 4.

Table 4: CPU Elapsed Time for PCT

Size	DM	SM	RM	SM/DM	RM/DM
200*200	3.659	0.468	0.337	0.128	0.092
199*199	3.639	0.451	0.273	0.124	0.075

DM=Direct Method, SM=Symmetric Method, RM=Relative Prime Method

The results from Table 4 show significant reductions in average CPU elapsed time for PCT. We can observe that both SM and RM based transform are effective compared with direct transform. FPCT (RM based transform) performs best. For odd and even number size image, FPCT only take 8% and 9% time compared with PCT respectively. Computation time is significantly reduced for PCT and similar with the theoretical analysis in Section 3.

5 Conclusion

In this paper, Fast Polar Cosine Transform is proposed. Based on previous work, the proposed method novelly employs relative prime points to accelerate the speed. Based on the number theory theorem, large number of trigonometric functions are saved when calculating points in the same angle. FPCT is about 12.5 and 11 times faster than PCT for odd and even number size image respectively. Comprehensive experimental results are given to demonstrate the effectiveness. Wide range of real world applications that need Fast Polar Cosine Transform will benefit from this method.

Acknowledgment

This work was supported in part by Grant-in-Aid (No.21500181) for Scientific Research and a grant of Knowledge Cluster Initiative 2nd stage by the Ministry of Education, Culture, Sports, Science and Technology (MEXT), Japan.

References

[1] P. Yap, X. Jiang and A.C. Kot, "Two Dimensional Polar Harmonic Transforms for Invariant Image Representation," IEEE Trans. PAMI, 15th May, 2009.

[2] Z. Yang and S. Kamata, "Fast Polar Harmonic Transforms," 11th International Conference on Control ,Automation ,Robotics and Vision, 2010.

[3] C.S. Burrus and T.W. Parks, "DFT/FFT and Convolution Algorithms and Implementation," John Wiley & Sons,1985.

[4] R. Guy, "Unsolved Problems in Number Theory," 3rd ed., Springer Press,2004.

[5] J. Dence and T. Dence, "Elements of the Theory of Numbers," Harcourt Academic Press,1999.

[6] G. Hardy and E. Wright, "An Introduction to the Theory of Numbers," 6th ed., Oxford University Press,2008.