Study on Algorithm for Panoramic Image Basing on High Sensitivity and High Resolution Panoramic Surveillance Camera

Zhenhai Zhang
School of Mechatronics Engineering
Beijing Institute of Technology
5 South Zhongguancun Street
Haidian District
Beijing 100081, P.R. China
zhzhang@bit.edu.cn

Kejie Li
School of Mechatronics Engineering
Beijing Institute of Technology
5 South Zhongguancun Street
Haidian District
Beijing 100081, P.R. China
likj@bit.edu.cn

Abstract

A single panoramic annular lens optical system and high-sensitivity, high-resolution CCD sensors form the basis of a 360° panoramic night vision image processing hardware platform. The unwrapped and correcting algorithm based on Coordinate Rotation Digital Computer (CORDIC) and bilinear interpolation algorithm was presented in this paper, with the purpose of processing dynamic panoramic annular image. The night vision image enhancement algorithm, based on adaptive piecewise linear gray transformation (APLGT) and Laplacian of Gaussian (LOG) edge detection, were given. An original annular panoramic image captured by panoramic annular lens (PAL) can be unwrapped and corrected to conventional rectangular image without distortion, which is much more coincident with people's vision. APLGT algorithm can be adaptively truncate the image histogram on both ends to obtain a smaller dynamic range so as to enhance the contrast of the night vision image. LOG algorithm can be propitious to find and detect dim small targets in night vision circumstance. The algorithm for panoramic image processing is modeled by VHDL and implemented in FPGA. The experimental results show that the proposed panoramic image algorithm for unwrapped and distortion correction has the lower computation complexity and the architecture for dynamic panoramic image processing has lower hardware cost and power consumption.

1. Introduction

Panoramic imaging techniques can image a full hemispherical, or even greater, field of view by means of special imaging devices. A panorama is an image having a wide horizontal field of view, up to a full 360°. Conventional video cameras have a limited field of view, usually smaller than the human field of view. The panoramic image plays more and more important roles; it is extensively applied in fields such as robot navigation, virtual reality, surveillance, computer vision, and visualization. Processing speed and precision are the critical features of a dynamic panoramic image algorithm, especially for real-time and high-quality image applications. Hence, a great deal of significant research has been done on panoramic image algorithms in recent years.

There are many different panoramic image processing algorithms based on different panoramic image acquisition methods. Image stitching algorithms are a common method. The images are obtained by multiple cameras or one rotating camera. Two images capture different portions of the same scene with an overlapping region in both images. The aligned images are then projected onto a plane or cylinder to stitch into a panoramic image. The disadvantages of image stitching algorithms are that additional cameras increase the cost and that the need for moving parts decreases the robustness of the system. Another panoramic image algorithm is a distortion correcting algorithm. The panoramic image is taken using a single camera and fish-eye lens or special mirror (e.g., conical, spherical, hyperboloidal). Through the special mirror or fish-eye lens, a single image encompasses the entire scene. The shortcoming is that the geometric structure of the optical system is complicated [1].

In this paper, we propose the algorithm for unwrapped and distortion correction basing on CORDIC and bilinear interpolation algorithm for panoramic image, by utilizing of FPGA as a video processor. The original panoramic images in this article are obtained by panoramic annular lens (PAL). Panoramic Annular Lens' main advantage to other omni-directional monitoring systems is that it is a cheap, small, compact device with no external spherical, conical or para-boloidal reflecting mirror as in other panoramic optical devices.

978-1-4799-0703-8/13/$31.00 ©2013 IEEE

2. Structure and principle of panoramic annular lens

The geometric structure of the PAL is illustrated in Figure 1. The PAL is a single element lens made from a high index of refraction glass with certain portions of the lens coated with a mirrored surface. PAL contains two reflecting surfaces and two refracting surfaces and relies upon both reflection and refraction in forming an image. And it can provide a panoramic view of the entire 360 degrees surrounding the optical axis. The lens is unique in that it images a three-dimensional object field onto a two dimensional image plane. A "normal" lens is capable of only imaging two-dimensional object space onto the image plane. The width of the annular image of the cylinder of vision will correspond to the viewing angle θ , which equals to the sum of α and β in the direction of the axis. The viewing angle θ covers the entire 360 degrees surrounding the optical axis. This combination of reflecting surfaces and refracting surfaces provides the PAL with a field of view extending from approximately $-\beta$ degrees to $+\alpha$ degrees from the optical axis, which β is about 12 degrees and α is about 33 degrees. As shown in Figure.1, it depicts a ray diagram for the PAL, showing the path a ray will traverse inside the lens. Rays A and B show the minimum and maximum ray acceptance angles that are captured onto the PAL image plane. Any ray originating from outside the θ degree field of view will not form a part of the image. And A' and B' are image points of A and B according to the beam path [2].

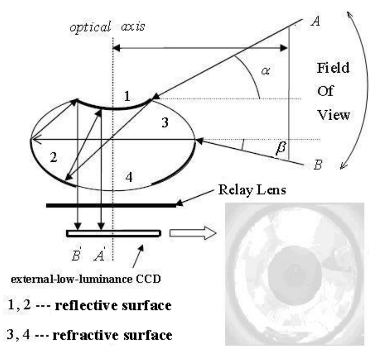

Figure 1: Structural principle sketch of panoramic annular lens optical system

3. Panoramic annular image processing algorithm

3.1 Analysis of panoramic image unwrapped and distortion correction algorithm

The resulting annular panoramic image is shown in Figure 1 (bottom right). These panoramic annular images (i.e., "donut" images) suffer from wide distortion and are not easy for a human observer to view. To improve the visual comprehension, a dynamic panoramic image processing algorithm is proposed in this article. The "donut" image captured by the PAL is unwrapped and corrected to fit within a conventional rectangular image without distortion. The algorithm is based on a CORDIC and bilinear interpolation algorithm and considers the characteristics of an annular panoramic image. The algorithm is written in VHDL and implemented on an FPGA video processor to make use of the powerful computational capability and good programmability of FPGA hardware. Figure 2 shows the geometry used in the unwrapping algorithm. The "donut" image is mapped onto the output frame using a polar to rectangular coordinate transformation. According to Figure 2, unwrapping the "donut" image into a conventional rectangular panoramic view results in a panoramic picture with a maximum height of $R-r$. The length of the unwrapped image is $2\pi R$, which equals the perimeter of the outer circle, that is, the full 360° view of the scene. The size of the unwrapped picture is $2\pi R\times(R-r)$.

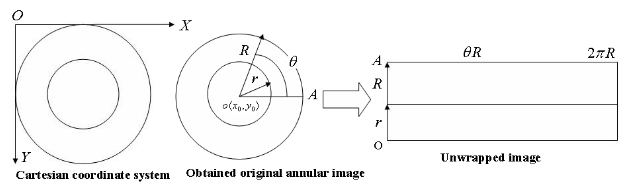

Figure 2: Principle of the panoramic annular image unwrapping and distortion correcting

3.2 Analysis of CORDIC algorithm

After analyzing the principle of panoramic image processing algorithm, the complexity of its computation is considered as the key problem of the implementation of dynamic panoramic visual system. Because the panoramic image algorithm uses multipliers and look-up tables containing trigonometric functions, such as sine, cosine function, this results in complicated hardware, insufficient speed and low precision. And so, we proposed the algorithm based on CORDIC for panoramic image unwrapped and distortion correction in order to implement by means of FPGA as a processor. By utilizing CORDIC algorithm, it is very easy to implement for computing sine, cosine trigonometric functions and coordinate transformation from polar to Cartesian.

CORDIC (Coordinate Rotation Digital Computer), advanced by J.D Volder in 1959, is a method for computing elementary functions using minimal hardware such as shifts, adds, subs and compares. CORDIC works by rotating the coordinate system through constant angles until the angle is reduces to zero. The CORDIC algorithm performs a planar rotation. Graphically, planar rotation means transforming a vector (X_i,Y_i) into a new vector (X_j, Y_j) , as shown in Figure.3. The principle is described as the following equations. Using a matrix form, a planar rotation for a vector of (X_i, Y_i) is defined as

$$\begin{bmatrix} X_j \\ Y_j \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \end{bmatrix} \quad (1)$$

The angle rotation can be executed in several steps, using an iterative process. Each step completes a small part of the rotation. Many steps will compose one planar rotation. A single step is defined by the following equation:

$$\begin{bmatrix} X_{n+1} \\ Y_{n+1} \end{bmatrix} = \begin{bmatrix} \cos\theta_n & -\sin\theta_n \\ \sin\theta_n & \cos\theta_n \end{bmatrix} \begin{bmatrix} X_n \\ Y_n \end{bmatrix} \quad (2)$$

Equation 2 can be modified by eliminating the $\cos\theta_n$ factor.

$$\begin{bmatrix} X_{n+1} \\ Y_{n+1} \end{bmatrix} = \cos\theta_n \begin{bmatrix} 1 & -\tan\theta_n \\ \tan\theta_n & 1 \end{bmatrix} \begin{bmatrix} X_n \\ Y_n \end{bmatrix} \quad (3)$$

If we restrict the rotation angle steps so that the tangent of a step is a power of 2, multiplying or dividing by the tangent can be implemented using a simple shift operation and additional multipliers can be eliminated. The angle for each step is given by

$$\theta_n = \arctan\left(\frac{1}{2^n}\right) \quad (4)$$

All summed iteration-angles must equal the rotation angle

$$\sum_{n=0}^{\infty} S_n \theta_n = \theta \quad (5)$$

Where

$$S_n = \{-1; +1\} \quad (6)$$

This results in the following equation for $\tan\theta_n$

$$\tan\theta_n = S_n 2^{-n} \quad (7)$$

Combining equation 3 and 7 results in

$$\begin{bmatrix} X_{n+1} \\ Y_{n+1} \end{bmatrix} = \cos\theta_n \begin{bmatrix} 1 & -S_n 2^{-n} \\ S_n 2^{-n} & 1 \end{bmatrix} \begin{bmatrix} X_n \\ Y_n \end{bmatrix} \quad (8)$$

Besides for the $\cos\theta_n$ coefficient, the algorithm has been reduced to a few simple shifts and additions. The coefficient can be eliminated by pre-computing the final result. The first step is to rewrite the coefficient.

$$\cos\theta_n = \cos\left(\arctan\left(\frac{1}{2^n}\right)\right) \quad (9)$$

The second step is to compute equation 9 for all values of 'n' and multiplying the results, which we will refer to as K.

$$K = \frac{1}{P} = \prod_{n=0}^{\infty} \cos\left(\arctan\left(\frac{1}{2^n}\right)\right) \approx 0.607253 \quad (10)$$

K is constant for all initial vectors and for all values of the rotation angle, and it is normally referred to as the congregate constant. The derivative P is the radius factor, which equals 1.64676 in circular operation mode [3, 4].

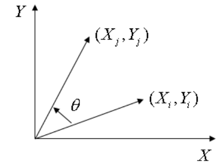

Figure 3: sketch map of planar rotation based on CORDIC

3.3 Bilinear interpolation algorithm

Pixels in unwrapped rectangular image can be constructed from those in original panoramic annular image. Coordinate of any pixel in original panoramic annular image can be computed by means of CORDIC. Result coordinate of unwrapped rectangular image is used to get a pixel value from the original image. This pixel value is used to fill the corresponding pixel in unwrapped rectangular image. Since $\sin\theta$ and $\cos\theta$ are fractions and less than 1, the calculated coordinate may not be an integer. The pixels in different images can't be mapped one-to-one. It is necessary to do interpolation to get correct value for pixel of unwrapped rectangular image. There are some common processing methods, such as nearest neighbor interpolation, bilinear interpolation and bicubic interpolation. We choose the bilinear interpolation in the implementation because of its high precision and relatively less complexity.

Fill-in algorithm can interpolate one or several gray values in the space between adjacent scan lines where there are no sampling data. The gray values of pixels of no sampling data may be calculated by fill-in algorithm. Actually, bilinear interpolation is a smoothing filter. And the principle of bilinear interpolation algorithm is described as the following equations:

$$f(x,y) = (j+1-y)f(x,j) + (y-j)f(x,j+1) \quad (11)$$

$$f(x,j+1) = (x-i)f(i+1,j+1) + (i+1-x)f(i,j+1) \quad (12)$$

$$f(x,j) = (x-i)f(i+1,j) + (i+1-x)f(i,j) \quad (13)$$

$$i = \left(x \times \frac{InputWidth}{OutputWidth}\right) \quad (14)$$

$$j = \left(y \times \frac{InputHeight}{OutputHeight}\right) \quad (15)$$

where $0 \le x < OutputWidth, 0 \le y < OutputHeight$. $OutputWidth$ and $OutputHeight$ are the width and height of output image, respectively. $InputWidth$ and $InputHeight$ are the width and height of input image, respectively. $f(x,y)$ is the gray value of pixel (x,y) . $f(i,j)$, $f(i+1,j)$, $f(i,j+1)$ and

$f(i+1, j+1)$ are the sample values of four neighbor points around pixel (x, y).

Equations (11) are simple and easy to be realized with FPGA hardware and the interpolation speed is very fast.

4. Night vision image enhancement algorithm

The image enhancement is of great importance in the area of night vision technology. The main objective of image enhancement is to process an image in order to make the images more suitable for human visual system. Firstly the spatial domain approaches are based on direct manipulation of pixels in an image, APLGT histogram enhancement algorithm, was discussed. And then LOG edge detection algorithm was discussed.

4.1 APLGT histogram enhancement algorithm

The APLGT histogram enhancement algorithm is a pixel histogram transformation function that enhances the contrast of image. The dynamic range of the histogram components are concentrated on the low side of the gray scale in a night vision image, and, generally, the dynamic range of night vision is narrow. A piecewise linear gray transformation allows that narrow dynamic range to be expanded to use the entire range of the image. The pixel values within the narrow dynamic range are displayed with marginal loss of quality, while information in the other parts of the image is discarded, increasing the contrast of the image. The diagram of the pixel transformation function is shown in Figure 4.

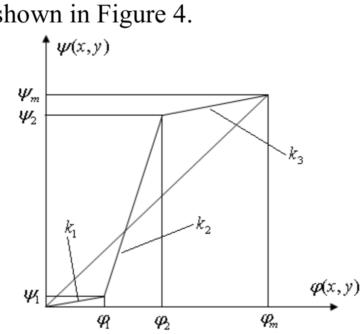

Figure 4: sketch map of piecewise linear gray transformation

According to Figure 4, the expression of piecewise linear gray transformation is defined as

$$\psi(x,y) = \begin{cases} k_1\varphi(x,y), 0 < \varphi(x,y) < \varphi_1 \\ k_2\left[\varphi(x,y) - \varphi_1\right] + \psi_1, \varphi_1 \le \varphi(x,y) < \varphi_2 \\ k_3\left[\varphi(x,y) - \varphi_2\right] + \psi_2, \varphi_2 \le \varphi(x,y) < \varphi_m \end{cases} \quad (16)$$

Where $k_1 = \psi_1 / \varphi_1, k_2 = (\psi_2 - \psi_1)/(\varphi_2 - \varphi_1)$ and $k_3 = (\psi_m - \psi_2)/(\varphi_m - \varphi_2)$. Function $\varphi(x,y)$ is mapped to function $\psi_m(x,y)$ in expression (16).

The above transformation method is just an example according to piecewise linear gray transformation for three line segments. In practice, the whole gray range can be divided into any segment of transformation section according to practice application.

The key idea of the piecewise linear transformation algorithm is to first truncate the image histogram on both ends to obtain a smaller dynamic range for the image pixel values and then to spread out the pixel values in adaptive range. The APLGT histogram enhancement algorithm is the one which have the adaptive property and be good for a large number of images with the characteristic of different image. The adaptive piecewise linear transformation with three line segments based on clipping principle is shown in Figure 5. It can transform the gray scale below the begin point into the same gray level 0 and those above the end point into one gray level 255.

Let $\{a_k, n_k\}$ be defined as follows. a_k is the k th gray level and n_k is the number of pixels that have a_k gray level. Let a_0 denote the pixel gray of the most frequency and n_0 denote the number of times that the gray level a_0 appears in image. And then let n_T be defined as follows: $n_T = n_0 \cdot 10\%$.

Between the linear region $[0, a_0]$, it is necessary to exist $\{a_L, n_L\}$, all the gray level in region $[0, a_L]$ satisfy the equation $n_i < n_T$,where $i \in [0, a_L]$; Similarly, between the linear region $[a_0, 255]$, it is necessary to exist $\{a_R, n_R\}$, all the gray level in region $[a_R, 255]$ satisfy the equation $n_i < n_T$,where $i \in [a_R, 255]$.

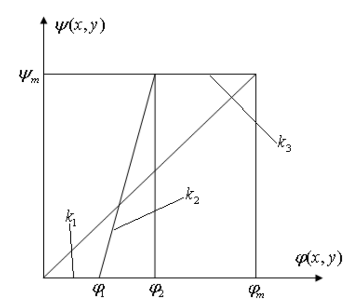

Figure 5: sketch map of adaptive piecewise linear transformation based on clipping principle

We can now deal with the modified transformation function as follows.

$$\psi(x,y) = \begin{cases} 0, \varphi(x,y) < a_L \\ \dfrac{1}{a_R - a_L} \cdot (\varphi(x,y) - a_L), a_L \le \varphi(x,y) \le a_R \\ 1, \varphi(x,y) > a_R \end{cases} \quad (17)$$

Where function $\varphi(x,y)$ denotes the gray value of original image. And transformation function $\psi(x,y)$ is the gray value of enhancement image by means of piecewise linear transformation. Transform intensity values $[0,a_L]$ to 0, $[a_R,255]$ to 255,and $[a_L,a_R]$ to $[0,255]$ using piecewise linear stretching.

The transformation function is approximated by a piecewise linear function, which to transform the original pixel values to enhancement ones for the displayed image and increase the slope of the grayscale function to realize contrast compensation.

The APLGT histogram enhancement algorithm is more flexible and can be realized simply and has high performance in enhancing the contrast of an image. As shown in Figure 6, the left half area is the original night vision image and the right half area is the enhancement image by means of APLGT histogram enhancement algorithm.

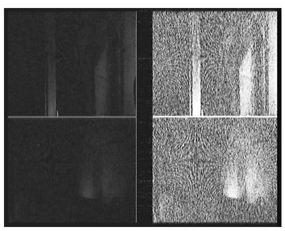

Figure 6: night vision image and adaptive piecewise linear gray transformation histogram enhancement image

4.2 LOG edge detection algorithm

With regard to the night vision image, it has some inherence characteristics of low contrast between targets and background, blurring edges and bigger noise than other visible images. In this paper we use LOG edge detection algorithm.The LOG operator is actually a combination of a low pass filter and a high pass filter. In practice a Gaussian is employed to smooth the image before the Laplacian is applied.

A two dimensional Gaussian function $H_\sigma(x,y)$ is as follows:

$$H_\sigma(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (18)$$

Where σ is the standard deviation of the Gaussian filter. With Fourier

$$h_\sigma(u,v) = e^{-\frac{1}{2\sigma^2}(u^2+v^2)} \quad (19)$$

The input image is smoothed by a Gaussian, $H_\sigma(x,y) * f(x,y)$. Where * denotes convolution, $f(x,y)$ is the input original image.

A laplacian operation is then applied to the convolution. Since

$$\nabla^2[H_\sigma(x,y) * f(x,y)] = \nabla^2 H_\sigma(x,y) * f(x,y) \quad (20)$$

The two operations can be combined into one filter, the LOG operator, which is defined by

$$\nabla^2 H_\sigma(x,y) = \frac{1}{\pi\sigma^4}\left(1-\frac{x^2+y^2}{2\sigma^2}\right) e^{-\frac{1}{2\sigma^2}(x^2+y^2)} \quad (21)$$

And then, we can get the Fourier transform of the LOG operator.

$$F[\nabla^2 H_\sigma](u,v) = -2\pi^2(u^2+v^2) e^{-\frac{1}{2\sigma^2}(u^2+v^2)} \quad (22)$$

Where F denotes the Fourier transform and $f(x,y)$ and $F(u,v)$ is a Fourier transform pair.

Rosenfeld and Kak showed that the simplest isotropic derivative operator is the Laplacian, which, for input image function $f(x,y)$ of two variables, is defined as

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \quad (23)$$

We can use different approximations for make proper model for laplacian operator.

$$Laplacian \cdot Mask_1 = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} \quad (24)$$

$$Laplacian \cdot Mask_2 = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix} \quad (25)$$

We note from these masks that the above implementations are isotropic. Mask1 and Mask2 are isotropic for rotation increments of 90° and 45°, respectively.

According to values of σ variety of mask can be implemented. As shown in Figure 7, the left half area is the original image and the right half area is the enhancement image by means of LOG edge detection algorithm.

5. FPGA implementation of dynamic panoramic image processing system and experimental results

Nowadays, FPGA technology is applied widely to implement logic circuits because it is configurable and parallelism can be exploited to achieve high speed. An FPGA can be programmed with many different hardware definition languages, such as VHDL, Verilog HDL, et al. And for this paper's image processing system the language of choice was VHDL. We use VHDL to describe the logic function of system and image processing algorithm with pipeline structure. Shown in Figure 8 is a block diagram of dynamic panoramic image processing system based on FPGA.

The video stream from the panoramic annular lens camera is sent to the video processor system for dynamic panoramic image processing. The proposed image processing system is constructed with panoramic annular lens video camera and FPGA system constructed with PAL (Phase Alternating Line) video encoder/decoder chip. FPGA is LSI (Large Scale Integration) that has programmable and reconfigurable inner circuits. In this paper, the Logic Element number of FPGA is 1,000,000 system gates.

Figure 7: Original image and LOG edge detecting image

The dynamic panoramic processing algorithms are implemented on the FPGA video processor system automatically. The input images from panoramic annular lens video cameras are high sensitivity and high resolution. The obtained digital image data are stored in SRAM on the FPGA system and are processed in-time. All the experiments of the panoramic image processing algorithm are performed on a FPGA image processing system. The original panoramic annular input image is shown in Figure 9. Figure 10 is the unwrapped octagon direction rectangular image [5].

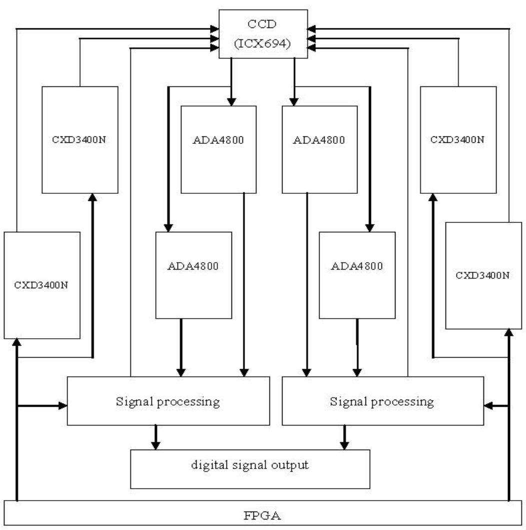

Figure 8: Block diagram of dynamic panoramic image processing hardware system

6. Conclusions

This article presented an image processing algorithm and its FPGA hardware implementation for dynamic panoramic annular images. The CORDIC algorithm is used to perform a coordinate transformation and image unwrapping, and a fill-in algorithm is applied to interpolate the image to eliminate distortion and enhance image quality. The practical performance and experimental results indicate that the speed of the coordinate transformation and image interpolation is rapid, the system architecture is simple, and the original panoramic annular image can be unwrapped and corrected into a conventional rectangular image without distortion and with significantly improved image quality.

Figure 9: Original Image taken by the panoramic annular lens

Figure 10: unwrapped octagon direction image

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant No. 61273346), the National Defense Major Fundamental Research Program of China (Grant No. C×××20110003), the National Defense Key Fundamental Research Program of China (Grant Nos. A×××20110005 and A×××20132010), the Specialized Research Fund for the Doctoral Program of Higher Education (Grant No. 20121101120009), the Excellent Young Scholars Research Fund of Beijing Institute of Technology (Grant No. 2012YG0203), the Program for the Fundamental Research of Beijing Institute of Technology (Grant Nos. 3020012211219 and 3020012251104), and the Program of Introducing Talents of Discipline to University, China (Grant No. B08043).

References

[1] M.P. Ghayalod, E.L. Hall, F.W. Reckellhoff et al, Line following using omnidirectional vision, Pnc. of SPIE Intelligent Robots and computer Vision conf, SPIE 2056, Boston, MA, 1993.

[2] Hui-quan Cheng, Wei-yong Yao, Guo-guang Yang et al, Design of an image transferring system used for panoramic annular imaging, Opto-Electronic Engineering 29(2), 16-19, 2000.

[3] Chun-Fu Lin and Sau-Gee Chen, A CORDIC algorithm with fast rotation prediction and small iteration number, Proc. of ISCS'98, vol. 5, pp. 229-232, 1998.

[4] R. Andraka, A Survey of CORDIC Algorithms for FPGAs, Proceedings of the 1998 ACM/SIGDA Sixth International Symposium on Field Programmable Gate Arrays, Monterey, CA, Feb, 1998.

[5] S. A. Guccione and D. Levi, XBI: A Java-based interface to FPGA hardware, Gonfigurable Computing Technology and its uses in High Performance Computing, DSP and Systems Engineering, Proc. SPIE Photonics East, J. Schewel (Ed), SPIE – the International Society for Optical Engineering. Bellingham, WA, November, 1998.