The Traffic Image Is Dehazed Based on the Multi–Scale Retinex Algorithm and Implementation in FPGA

Cui Zhe[1, a], Chao Li[2, b *], Jiaqi Meng[3, c]

[1]Shannxi Province Transportation Planning and Research Institute, Xi'an, China

[2]Highway College, Chang'an University, Xi'an, China

[3]CCCC First Highway Consultants CO. Ltd, Xi'an, China

[a]723648316@qq.com, [b]420837905@qq.com, [c]972937118@qq.com

Keywords: Retinex; image dehazing; FPGA

Abstract: This paper based on Retinex algorithm for image processing go fog, and Retinex algorithm for some optimization in FPGA, in slightly affect image quality while significantly reducing the FPGA resources. Then on the successful implementation of the algorithm in FPGA, and with the effect on matlab found to compare with the original fog effect FPGA matlab results are basically the same.

Introduction

In the scenarios of "Safe City" project, banking, museology, hotel, office building, residence community, safe village, campus, harbor, highway, street, etc., conventional camera is difficult to meet the need of 24 hours continuous monitoring due to high performance demand, so low illumination cameras become a priority. In the mentioned scenarios, there is neither enough illumination nor possibility to install lighting facility on a large scale. In this case, high performance and low illumination cameras are required to ensure monitoring quality and simplify system framework, which leads to good reliability and low cost.

Along with application growth of low illumination camera, the technology is in constant progress. There were types of low illumination cameras: electronic day and night camera, slow shutter camera, professional super low illumination camera (usually ICR, mechanical color to black type).

Electronic day and night camera uses electronic circuit to switch colored images into monochrome in order to improve its ISO in low illumination. Slow shutter camera, also named frame accumulation camera, can increase image brightness by extending exposure time. When its slow shutter is on, the output images are not real time. As long as the camera shutter decelerates more than 4 times, image ghosting can be obviously seen on the monitor[1]. Slow shutter camera enhances image sharpness by sacrificing instantaneity not improving device hardware performance. Professional super low illumination camera (usually ICR, mechanical color to black type) enhances image quality under low illumination by adopting high-performance CCD, professional low illumination circuit design, ICR filter switcher, etc., which guarantees real time and vivid image output. Demand for the product function is more practical to implement. For instance, defocus occurs after transformation from color to black and frequent day and night switch happens when low illumination camera cooperates with infrared light. However, the 3 cameras are monochrome without color information, which influences practicability[2].

Along with technology development, more and more new theories spring to solve the problems. Color constancy means that in different environments, human eyes sensing to color is constant in some certain range. Color constancy theory simulates the unique function of human visual system to process images with high recognition to get better visual effect[3].

Land and McCann proposed the color constancy algorithm representing human visual system, named as Retinex thoery after the combination of retina and cortex[4]. However, due to uneven illuminancy of images, Land raised the random walk algorithm, which is too complicated to achieve. Then Jobson et al. put forward the classic center/around theory single-scale Retinex(SSR)[5], which solve the complicatedness problem. But other problem came up due to scale of SSR, some scholars suggested multi-scale Retinex(MSR)[6] to solve the scale issue. Meanwhile chromatic aberration of MSR is serious, multi-scale recovered color Retinex(MSRCR) was raised. MSRCR is good at decrease chromatic aberration, but still complicated in achieving the real time monitoring.

Implementation of low illumination algorithms are currently using high performance digital signal processor(DSP), but it is also difficult to achieve real-time. Recently development of microelectronics technology and manufacture technology of ultra large scale integrated circuit, especially Field Programmable Gate Array(FPGA), provided new thoughts and method for improving performance of image processing system[7]. FPGA is one of the most highly integrated circuits. Users can reconfigure the internal logic module and I/O module of FPGA to implement function needed. FPGA is initialized through putting code into chips and electrifying, while it can be coded online to restructure system. The FPGA is rich in logical units, and easy to achieve a variety of circuit design and perform complicated operations, so designers just need to modify inner logical functions for different image process requirements by software. Meanwhile, highly integrated FPGA makes image panel simpler, layout more compact. This paper proposed the low illumination design solution based on FPGA, aiming to achieve low illumination algorithm and real time through using FPGA as process chips, Hardware parallel algorithm and pipelined structure[8].

MSR algorithm and optimization

MSR algorithm is evolved from Single-Scale Retinex algorithm, and the formula to acquire SSR images is shown below[9]:

$$R_i(x, y, c) = \log\{I_i(x, y)\} - \log\{F(x, y, c) \otimes I_i(x, y)\} \tag{1}$$

Ri(x,y,c) is noted as the channel i value of output image, Ii(x,y) as the channel I value of original image, representing convolution, and F(x,y,c) as Gaussian template, which is calculated below:

$$F(x, y, c) = Ke^{-(x^2+y^2)/c^2} \tag{2}$$

K is noted as normalized factor, and this factor should meet the condition below:

$$\iint F(x, y, c)dxdy = 1 \tag{3}$$

In formula (2), c represents scale of different Gaussian templates, namely the diameter of Gaussian template. And final MSR output is the weighted sum of several SSR output at different scales, which is formulated as:

$$R_{M_i}(x, y, w, c) = \sum_{n=1}^{N} w_n R_i(x, y, c_n)$$

(4)

RMi(x,y) is noted as the MSR output of channel i, $c \in \{c1, c2 \ldots cN\}$ represents different scales, and $w \in \{w1, w2 \ldots wN\}$ represents weight of different scales and it meets $\sum_{n=1}^{N} w_n$. Theoretically scale changes along actual conditions, most of time 3 different scales are required, 15, 80, and 250 pixels for instance, and weight can be the same.

However, color image looks grey due to low contrast after ordinary MSR process, one more step need to be done[10]:

$$R'_{M_i}(x, y, w, c, C) =$$
$$R_{M_i}(x, y, w, c) * I'_i(x, y, C)$$

(5)

Where $I'_i(x, y, C)$ is:

$$I'_i(x, y, C) = \log\left[1 + C * \frac{I_i(x, y)}{\sum_{i=1}^{3} I_i(x, y)}\right]$$

(6)

Here the initial log form log(x) is replaced by log(1+x) to make sure all the results positive which is favorable for FPGA implementation later. And C value comes from experience[11], which will not make much different to the image effect, so it is set at 125.

Considering that FPGA is resource limited and hard to run log, two hardware optimizations on initial MSR algorithm are made for easy FPGA implementation:

1. Cut the initial image twice, horizontal and vertical respectfully and get the quarter to process Gaussian Blur, which decrease the size of both the image and the Gaussian template, the resource consumption is cut by 3 quarters at least.

2. Approximate the log into arithmetic in formula (4) and (5), and combine them as:

$$R'_{M_i}(x, y, w, c, C) =$$
$$C * \frac{I_i(x, y)}{\sum_{n=1}^{N} \{w_n [F(x, y, c_n) \otimes I_i(x, y)]\}}$$

(7)

After that, the operations are reduced and all the results are positive.

And with MSR processing, just linear Enhancement to the histogram can achieve image dehaze effect.

FPGA implementation

Due to FPGA properties of ASIC fusion and processor based system, it can be programmed in high speed parallel scene to make high speed parallel data processing, complex calculation, mass data processing. By the development of IC craft, FPGA becomes more and more important in digitals with its advantages like good performance, cost, stability and convenience in maintenance and upgrade. FPGA outruns PC software and DSP/ARM on efficiency and bandwidth through

hardware acceleration in digital images processing. Meanwhile FPGA has succeeded in ARM and logical embedded system which is advantageous to parallel computing and complex control.

This paper uses FPGA of Altera Cyclone IV generation, type is EP4CE22E17C8N, this chip has 22000 LE, 132 embedded multipliers, 594KB RAM, 154 user IO, etc. which is enough for Retinex implementation. The software is Quartus II 13.0, programmed through Verilog HDL, using logic circuit to implement and verify hardware acceleration of Retinex image process algorithm. USB communication mechanism is applied to complete particular images transmits for process and analysis. And the framework of FPGA hardware acceleration image processing is shown in figure 1:
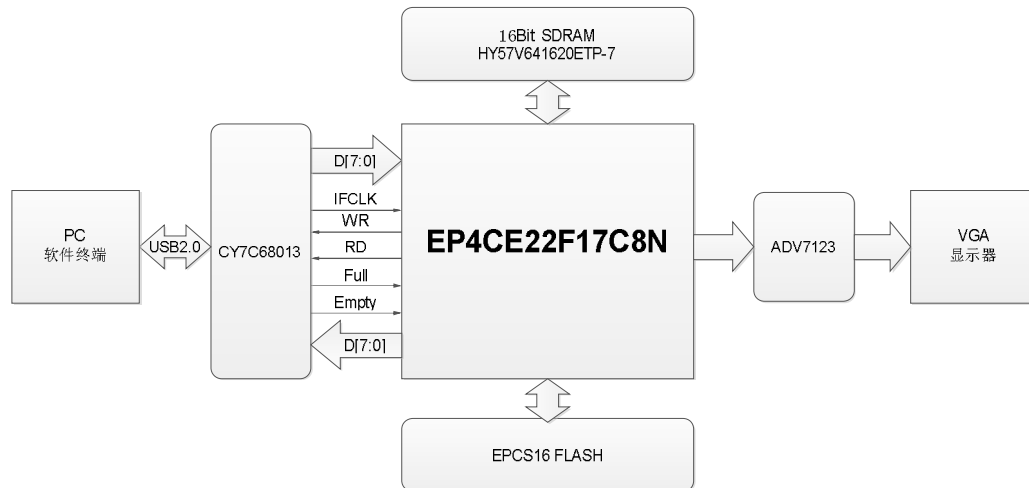

Fig.1 FPGA hardware framework of Retinex algorithm

Shown in Fig. 1, through PC software terminal and USB 2.0 protocol, the particular images would be sent to FPGA. The hardware uses proven USB 2.0 controllers of Cypress to complete data transmit-receive. FPGA through the ranks of the cache and the parallel hardware acceleration implements Retinex algorithm logical schema. After image processing, FPGA passes images to PC by USB and to VGA for display by SDRAM in order to observe, save, and analyze.

It is very appropriate to implement Retinex on image real time processing with Verilog HDL programming and rich multipliers and RAM in FPGA. In this paper, through Shift-RAM within the FPGA, complete image cache operations on a certain scale. This algorithm has only several clock periods, which means even nothing comparing to PC software. Shown as Fig. 2, it is simulation wave form result using Modelsim for Shif-RAM and single channel Retinex. The experiment verifies that the results fit the design algorithm and logic application can be achieved.
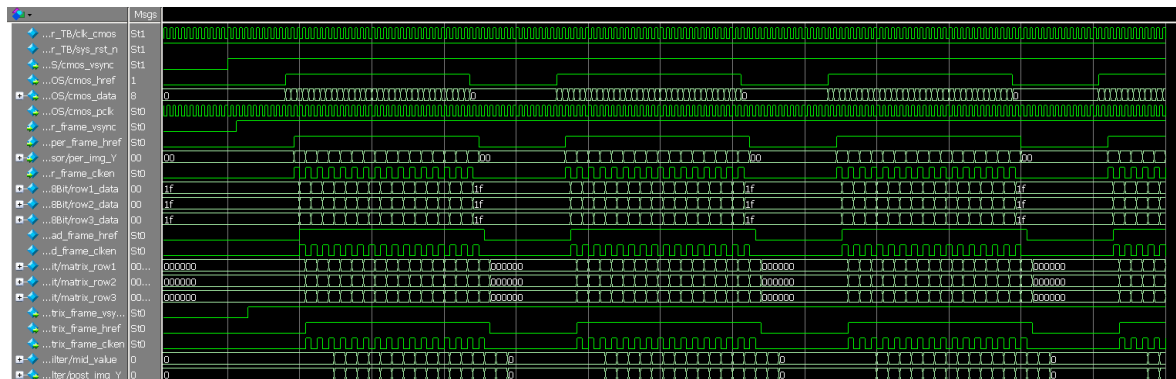

Fig.2 Modelsim Retinex Algorithm Simulation Test Wave

Result and analysis

Fig. 1, 2 and 3 are all processed by Retinex, (a) is original image, (b) is processed by matlab, (c) is processed by FPGA. It is can be seen that Rentinex has good dehaze effect and image quality damaged a little after FPGA implementation.
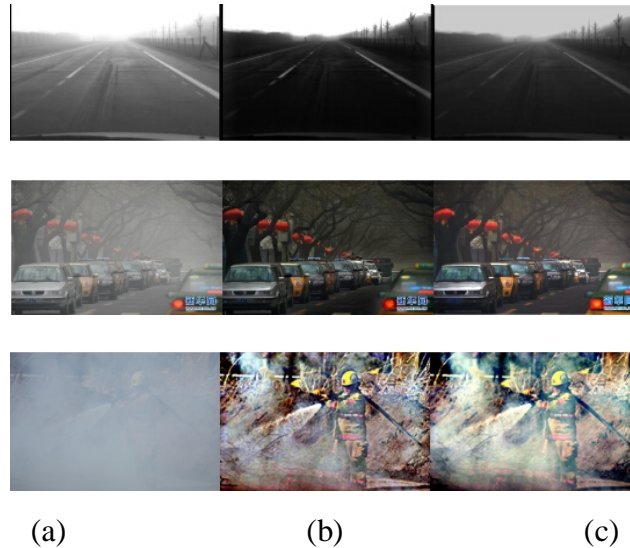

(a)   (b)   (c)

Fig.3 Effect images of Retinex Algorithm

Conclusion

This paper improved and optimized Retinex algorithm based on FPGA properties, speeded up the image processing while balanced the algorithm effect. And transplant the optimized algorithm to FPGA development kit of Altera which verified it doable and correctness. Compared to software implementation, FPGA significantly improved the processing efficiency of the algorithm and made cost lower and carriage easier which would fully satisfy the dehaze of images in transportation.

Reference

[1]. E. Land, J. McCann. Lightness and retinex theory. J. Journal of the Optical Society of America,1971,(01):7-21.

[2]. E. Land. An alternative technique for the computation of the designator in the retinex theory of color vision. J. Proceedings of the National Academy of Sciences(USA),1986,(10):3078-3080.

[3]. D. J. Jobson, Z. U. Rahman, G. A. Woodell. Properties ans performance of a center/surround Retinex. J. IEEE Transactions on Image Processing,1997,(03):451-462.

[4]. Z. U. Rahman, D. J. Jobson, G. A. Woodell. A multiscale retinex for color redition and dynamic range ccompression. on http://citeseerx.ist.psu.edu/viewdoc/summary?. doi =10.1.1.55.6939,2012.

[5]. Y. Liu, X. Gao. New algorithm for night Vision image enhancement processing. J. INFRARED AND LASER ENGINEERING 2006. S4:172-178.

[6]. Altera. Quartus II 13.0 Handbook. America, 2013.

[7]. X. Sun, A. Yu. VC++ in-depth Xiangjie. Electric Industrial Press. 2006.

[8]. Altera. Cyclone iv handbook. America, 2013.

[9]. Zhaiguo Fang, Huang Wei, Cheng Yun design [J] FPGA-based USB interface camera image data transmission system of the Sixth National signals and Intelligent Information Processing and Application Conference .2012.

[10] Liang A fast adaptive scale Retinex-based image enhancement algorithm [J]. Ninth National Optical Technology Symposium .2010.

[11]. Li,Yuecheng Zhang, Hong You etc. A multi-scale retinex implementation on FPGA for an outdoor application[J]. International Congress on Image and Signal Processing (CISP 2011).2011.