IMPLEMENTATION OF SOFTWARE ISP FOR THE QUALITY IMPROVEMENT OF AN IMAGE

Chitra S[1], Vadivel S[2], Dr. Roopa M[3]

[1]M.Tech VLSI design and embedded systems, Department of ECE, Dayananda Sagar College of Engineering, Bangalore, India
[2]Senior Technical Architect, Continental Automotive components private limited, Bangalore, India
[3]Professor, Department of ECE, Dayananda Sagar College of Engineering, Bangalore, India

---

Abstract - *The Image signal processor is a type of digital signal processor or media processor which is used for image processing in digital cameras or other devices. This ISP can receive the image data directly from the image sensor interface over a handshake mechanism or can read from memory. The output of the ISP can be given directly to another vision processor or can be used to store in memory. The image signal processor is used to convert and get more data of raw image data which is of Bayer format, received from the camera sensor. The paper is to develop the software which is dealing with the images captured by cameras.*

The software Image signal processor is developed which will be more flexible when compared to the actual image signal processor. K means algorithm is used, which uses averaging of the pixel values of corresponding images to increase the quality of an image. Resampling of the image is carried out at last block of ISP, resampling has two methods, Adaptive methods change depending on what they are interpolating and Non-adaptive methods treat all pixels equally. In this project, Lanczos Non-adaptive interpolation/resampling technique is used to resample the images, which uses 6 tap filtering techniques to interpolate and it is compared with the Bilinear interpolation technique to check the quality of an image this is carried out by calculating peak signal to noise ratio of images.

Key Words: Image signal processor, digital camera, resampling, 6tap filtering, PSNR.

1. INTRODUCTION

Image signal processor is a type of media processor which is used in all digital cameras. While capturing an image when the light passes through the sensors R, G and B images are formed depending on the wavelength of the light which can be called as Bayer format and these collections are called as pixels and these pixels are gone through mosaicing, demosaicing and many of other processes to form the clear picture of an image. The problem with RGB is that it is not the best mapping for representing visual perception. So to overcome this, YUV or YCbCr color space has come into picture which is more efficient in coding and reduces the bandwidth more than RGB bandwidth. Therefore most of the video cards use YUV, Y is called Luminance which is nothing but the grey level of an image, and UV is called chrominance which is the colored version of an image.

Most of the devices use an image in YUV color space to store an image and again decode back to RGB while displaying it. There are different formats of YUV they are, YUV444 with equal sample rate, YUV422 with the chrominance components are sampled horizontally at half the rate of the luminance, YUV420 with the chrominance components are sub-sampled at a factor of 2 in the vertical as well as the horizontal direction.

Image resampling is nothing but the process of interpolation. Image interpolation is nothing but an estimation of the values of unknown points, using known points i.e., estimation of unknown pixel values, using known pixel values. Image interpolation works in two directions and it tries to get the best approximation of pixel's intensity based on surrounding pixel values.

There are two different ways of interpolation they are, Adaptive and Non-adaptive algorithms.
The adaptive algorithm depends on what they interpolating and Non-adaptive algorithm treats all pixels equally. In this paper, we are considering two non-adaptive methods of interpolation.

2. DIFFERENT METHODOLOGY

2.1 Implementation of ISP for Reconfigurable processors

An image signal processor which is used for a CMOS image sensor consists of many complicated functions. So the full chain of ISP which functions for smart devices is can be implemented. Every function in the chain is fully converted to fixed-point arithmetic and no special function is used for easy porting to reconfigurable processors [1].

The fig1 below shows the flow of the vector processors, they are implemented. The image which is captured is first gone through normal mosacing and De-mosacing processes and white balancing after white balancing, Adaptive homogeneity-directed demosaicing (AHD). CIELab which is used as the color space in the original homogeneity-directed map is changed to YCbCr color space. After demosaicing, the color correction block finds color features and repairs color artifacts by using a linear stretch method for auto contrast. Gamma Correction with a reduced LUT(Lookup Table) is used and the gamma values between LUT entries are calculated by piecewise linear interpolation method.[7] for noise reduction algorithm, Modified Bilateral filter(BF) [6] is used and in the order of simplification two Gaussian filters are replaced by fixed-point binary threshold functions. The threshold values are determined by pre-calculating the Gaussian filter coefficient for pixel locations and pixel intensities. DoG(Difference of Gaussian) based LTI/CTI[8] is used for detail enhancement.

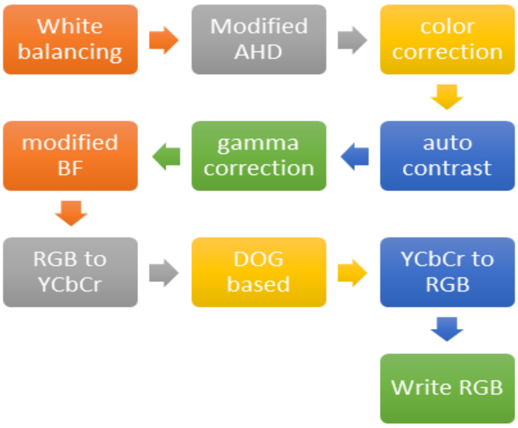

Fig-1: ISP full chain

2.2 Functional Blocks of ISP System

The front cameras which are used in the automotive system use a computer vision technique to detect the obstacles on road by the image which is captured. The ISP pipeline is tuned for computer vision applications to help the driver. This system is different in terms of inputs and outputs formats, processing needs, and algorithms for processing. The figure below shows the ISP pipeline which is proposed by considering the cameras of the automotive system.

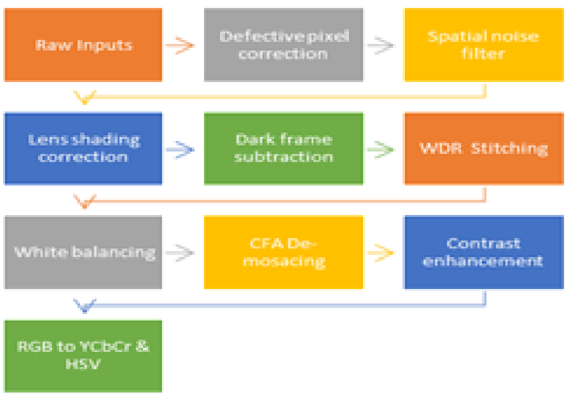

Fig-2: Functional block of ISP system

2.3 ISP using Non-linear Optimization

In this method an automatic Image quality (IQ) tuning using nonlinear optimization and automatic reference generation algorithms without being aware of their specific implementation. It was successfully implemented on different processing blocks such as noise reduction, demosaic, and sharpening.

Here for auto-tuning the image quality they are assumed the normal ISP block which consists of input correction, Bayer noise reduction, Demosaic, color correction, gamma correction, YUV noise reduction, and sharpening. They used steps,

Nonlinear Optimization, Parameter Abstraction, Automatic Reference Generator, and Fitness Calculation for auto tunning. Firstly, the images are get calibrated using reference images (different reference images for different modules are used).
Noise Reduction:

To get a different customer's IQ preference, the number of frames N can be adjusted.

$$Ifusion = 1/N \sum_{t=1}^{N} li(t)$$

For fine control, one can use blending weight of [0.0, 1.0] for 2 frames [2].

After this process, Demosaic is carried out where The Bayer image will be converted into RGB images.

Sharping

They used edge directed unsharp masking on the temporally averaged frame for that firstly, the gradients $G\{H,V\}$ using Scharr operator is used to calculate weight $w = |GH|/(|GH| + |GV|)$.

And the directional detail can be calculated as $Ddir = wDH \otimes Ifusion + (1 - w)DV \otimes Ifusion$
Where $DH = [-1,2,-1]T$ and $DV = DHT$
Non-directional detail is calculated as
$Dndir = Ifusion - Ifusion \otimes G\sigma USM$,
Where Gaussian low pass filter $G\sigma USM$ is configured as 9x9 with $\sigma USM = 2.5$.
Two components are blended as
$Iref = Ifusion + \alpha \cdot \alpha ndir(wndirDndir + (1 - wndir)Ddir)$
Where,

$wndir = exp(-min(GH,GV)2/\sigma ndir2)$ with $\sigma ndir = 0.5$,
$\alpha ndir = 1 - exp(-K \otimes |Dndir|/\sigma\alpha 2)$.

$\sigma\alpha$ is calculated as a mean sum with a standard deviation of $K \otimes |Dndir|$ at the flat region.

K is a 9x9 box filter. α controls an overall sharpness. [2]
Nonlinear optimization:

Global and local optimization are combined in the above approach to get a better result. In the first stage of the optimization, they will carry out global optimization. This stage will explore the high dimensional tuning parameter search space for good candidates, globally. At the second stage, local optimization is carried out to refine the candidates to further look for better fitness measurements. Artificial Bee Colony (ABC) is chosen for global optimization [9] and sublex method is chosen for local optimization [10].

2.4 Implementation of ISP on Raspberry Pi

The image signal processor can be implemented on raspberry pi, which is used to reduce the complexity of systems in a real-time application and to enhance the image which is captured in dark and low contrast images to identify the particular region of the image. This concept is used in MAV, which is used to capture videos and images through the camera module of the raspberry pi. The image which is captured by MAV consist of some noise from the atmospheric condition and it is necessary to remove that noise hence in this paper they are removing noise and applying Gaussian filtering to get a better image.

3. PROPOSED SOFTWARE ISP

In this paper, we are designing a software version of the image signal processor. The simplified block diagram of the proposed ISP is shown in fig 3.

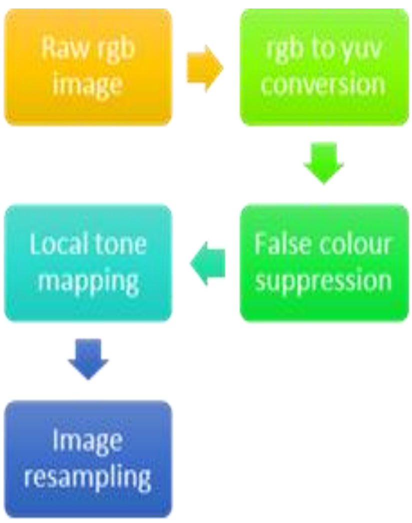

Fig-3: Proposed ISP block for the camera system

The raw RGB image from the camera sensor is split into R, G, and B and is converted to YUV image using the prescribed conversion formula.

These YUV images are gone through false color suppression and local tone mapping which are interconnected steps to increase the quality of an image. In false-color suppression, the color which is not perfectly determined by the image is get corrected by considering the neighboring pixel values. i.e., correction of the pixels by comparing with the next image pixels and produces a third picture of higher quality by interpolating. This image is then tone mapped to produce an image of high dynamic range in a medium or limited dynamic range, that is when the image is captured in the bright or dark time, then the quality of image reduces, so to get an image of high dynamic range we do tone mapping of an image.

There are different methods of tone mapping in that we are using K means algorithm for tone mapping which is comparatively easy and better method for tone mapping, This algorithm computes the luminance of an input i.e., radiance map which is generated from a set of lower dynamic range images with varying exposure settings, then an image is divided into several regions using k means algorithm. The display values are set automatically according to the mean value of each region.

The formula used here is,

$$T = 1/2 \sum_{i=1}^{N} \sum_{j=1}^{N} d(xi, xj)$$

$$1/2 \sum_{k=1}^{k} \sum_{i=k} \left(\sum_{j=k} d(xi, xj) + \sum_{j=k} d(xi, xj) \right)$$

Where,

d= Distance

Xi, Xj = pixel values of an image.

We have used the Lancoz method for interpolation which uses 6 tap filtering techniques for resampling of an image. The half pixels of Lancoz is generated by a trivial convolution procedure that is a single half-diagonal pixel depends on 36 integer pixels. Here we are incorporating this technique to maintain the quality of an image. This method of interpolation smoothly interpolates the pixels which can be approximated to sinc method to get a better image and it uses 6 tap FIR filters, whereas the bilinear interpolation method uses 2 tap FIR filter.

Peak signal to noise ratio gives the quality of an image when it is reconstructing. As the PSNR ratio increases, the image which is reconstructed is of better quality. The PSNR ratio is calculated by the formula,

$$PSNR = 10 \cdot \log_{10}\left(\frac{MAX_I^2}{MSE}\right) = 20 \cdot \log_{10}\left(\frac{MAX_I}{\sqrt{MSE}}\right) = 20 \cdot \log_{10}(MAX_I) - 10 \cdot \log_{10}(MSE)$$

4. RESULTS

Fig 4 shows the YUV image of the ISP block. This image is resampled using the Bilinear interpolation technique as well as lancoz interpolation technique and the peak signal to noise ratio of these two resampled imaged is calculated.

Fig-4: YUV image of resolution 860x430

The figure below shows the image which is captured at the bright and dark time and the output produced after tone mapping.

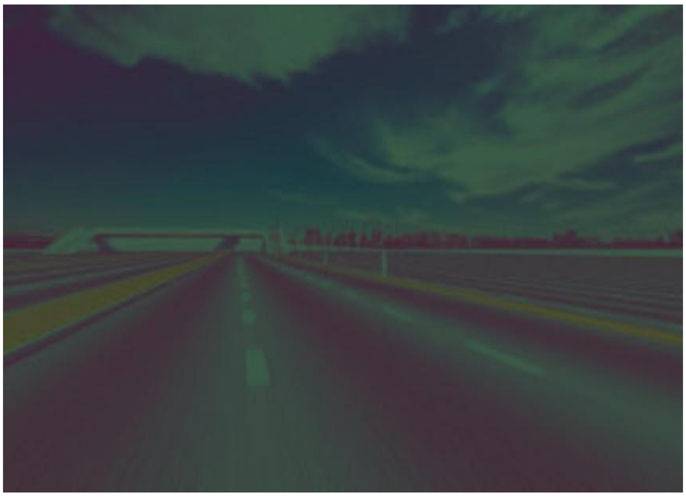

Fig-5: YUV image, captured at a dark time

Fig-6: YUV image, captured at a bright time

Fig -7: False color suppressed and tone mapped image.

Fig 8 shows the PSNR of an input image when it is resampled to different HD resolutions.

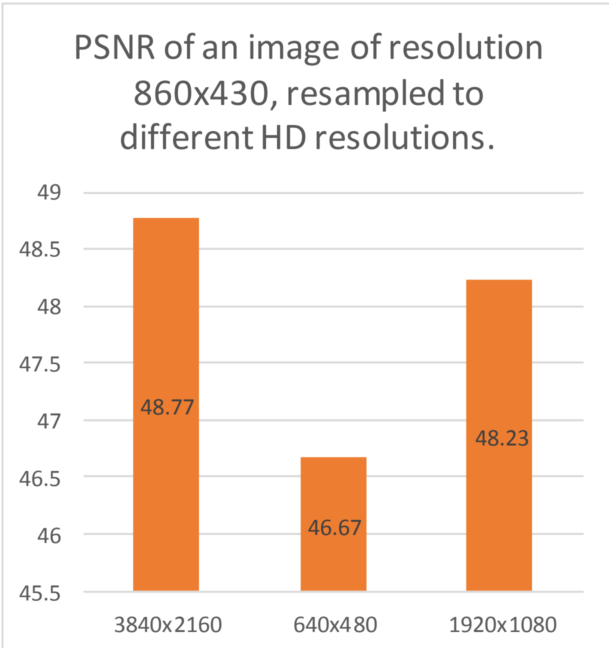

Fig-8: PSNR of lancoz interpolation image compared with bilinear interpolation image

5. CONCLUSION

In this paper, we have discussed various methodologies of implementing ISP, and the implementation of software ISP flow using K means algorithm for tone mapping has been used to get the improvised image. Lancoz method has been used to maintain the quality of an image and it is then compared with the image which was resampled using bilinear interpolation methodology, the peak signal to noise ratio of both images is computed wherein, their ratio yielded 48.77dB, 46.67dB and 48.23dB improvement for 3840x2160, 640x480 and 1920x1080 respectively which shows that the image quality has improved.

REFERENCES

[1] Seung-Hyun Choi, Junguk Cho, Yong-Min Tai, and Seong-Won Lee, "Implementation of an Image Signal Processor for Reconfigurable Processors", 2014 IEEE International Conference W.-K. Chen, *Linear Networks and Systems.* Belmont, Calif.: Wadsworth, pp. 123-135, 1993. (Book style)

[2] Mihir Mody, Niraj Nandan, Shashank Dabral, Hetul Sanghvi, Rajat Sagar, Zoran Nikolic, Kedar Chitnis, Rajasekhar Allu and Gang Hua , "Image Signal Processing for Front Camera based Automated Driver Assistance System" 2015 IEEE 5th International Conference

[3] Jun Nishimura, Timo Gerasimow, Rao Sushma, Aleksandar Sutic, Chyuan-Tyng Wu, Gilad Michael, "Automatic isp image quality tuning using nonlinear optimization",2018 IEEE.

[4] K.S.Shilpashree, Lokesha.H, Hadimani Shivkumar, "Implementation of Image Processing on Raspberry Pi", International Journal of Advanced Research in Computer and Communication Engineering Vol. 4, Issue 5, May 2015 .

[5] K. Hirakawa and T. W. Parks, "Adaptive homogeneity-directed demosaicing algorithm," IEEE Trans. Image Process., vol. 14, no. 3, pp.360–369, Mar. 2005

[6] C. Tomasi and R. Manduchi, "Bilateral Filtering for Gray and Color Images", Proc. 1998 IEEE Int'l Conf. Computer Vision, pp. 839-846, Jan. 1998

[7] D. Lee, R. Cheung, and J. Villasenor, "A Flexible Architecture for Precise Gamma Correction," IEEE Trans. Very Large Scale Integration (VLSI) Systems, vol. 15, no. 4, pp. 474-478, Apr. 2007.

[8] Yangang Wang, Xi Chen, Hua Han and Silong Peng, ಯ Video Luminance Transient Improvement Using Difference of Gaussian", In Proc. of 15th Asia-Pacific Conference on Communications (APCC 2009), pp. 249 – 253, Oct. 2009.

[9] Marjan Mernik, Shih-Hsi Liu, Dervis Karaboga, Matej Cˇ repinšek, "On clarifying misconceptions when comparing variants of the Artificial Bee Colony Algorithm by offering a new implementation," Vol. 291, pp.115-127, January 2015.

[10] Olsson, Per-Magnus, "Methods for Network Optimization and Parallel Derivative-free Optimization," Linköping Studies in Science and Technology: Dissertations, ISSN 0345-752.

[11] G.Senthilkumar1, K.Gopalakrishnan2, V. Sathish Kumar3 Embedded Image Capturing System Using Raspberry Pi System, Volume 3, Issue 2March–April 2014 Page 213.

[12] Sahani, M., Rout, S.K., Sharan, A.K., Dutta, S. real time color image enhancement with a high regard for restoration of skin color by using raspberry pi Communications and Signal Processing (ICCSP), 2014 International Conference IEEE.

[13] Y.Saahithi, E.Sai Spandana Reddy, P.Samskruthi Reddy, Advanced Embedded Security System With Image Capturing In SD Card, Volume No: 1(2014), Issue No: 12 (December).

[14] Ajinkya Patil1, Mrudang Shukla2, implementation Of Classroom Attendance System Based On Face Recognition In Class, IJAET, Vol. 7, Issue 3 July, 2014.

[15] Felix Heide, Markus Steinberger, Yun-Ta Tsai, Mushfiqur Rouf, Dawid Pajak, et al., "FlexISP: A flexible camera image processing framework," ACM Transactions on Graphics, vol. 33, no. 6, pp. 231, 2014.

[16] Jun Nishimura, Timo Gerasimow, Rao Sushma, Aleksandar Sutic, Chyuan-Tyng Wu, and Gilad Michael, "Automatic isp image quality tuning using nonlinear optimization," in Proceedings of IEEE International Conference on Image Processing, 2018, pp. 2471–2475.

[17] Samuel W Hasinoff, Dillon Sharlet, Ryan Geiss, Andrew Adams, Jonathan T Barron, Florian Kainz, Jiawen Chen, and Marc Levoy, "Burst photography for high dynamic range and low-light imaging on mobile cameras," ACM Transactions on Graphics, vol. 35, no. 6, pp. 192, 2016.

[18] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang, "Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising," IEEE Transactions on Image Processing, vol. 26, no. 7, pp. 3142–3155, 2017.

[19] Qifeng Chen, Jia Xu, and Vladlen Koltun, "Fast image processing with fully-convolutional networks," in Proceedings of IEEE International Conference on Computer Vision, 2017, vol. 9, pp. 2516–2525.

[20] Jan Erik Solem, Programming Computer Vision with Python, 2012.