https://doi.org/10.2352/ISSN.2470-1173.2021.10.IPAS-241

FPGA-based Fast Algorithm of Correcting Saturated Pixel in Image

Jun Fu, Chuanli Xiao, Jingyi Huang,Wenxue Peng, Xinpei Wang, Zhen Xu, Yanheng Wang,
Min Liu, Xuanqin Mou
Institute of Intelligent Computing and Data Communications, Xi'an Jiaotong University

Abstract

This paper proposes a novel method to correct saturated pixels in images. This method is based on the YCbCr color space and separately corrects the chrominance and the luminance of saturated pixels. In this algorithm, the saturated image is processed on the scan line, which is beneficial to the hardware implementation and the correction effect is good. Through the results of the joint simulation of MATLAB and Modelsim, it can be concluded that the hardware algorithm of this paper can use less resources to achieve fast correction. This paper uses Altera DE4 high-level development platform for hardware implementation. The calculation results show that high-speed image and video processing by FPGA is feasible and efficient, and it can be done frame by frame for high-definition video. It has broad practical application prospects.

Keywords: Saturated pixels correction, FPGA, Fast correcting while scanning

1.INTRODUCTION

Overexposure will result in saturated pixels in the image. In daily life, overexposure often occurs in the face part of the image, because the dynamic range of the scene is much wider than the dynamic range of the sensor. Overexposure may cause loss of detail and color distortion in highlight areas. In order to obtain an image without overexposure, we can reduce the exposure value. But lowering the exposure value may result in dark image and the details in the dark areas may be lost. Mishandling of saturated pixels can have serious negative effects on the image.

After inputting the image, our processing method judges whether the pixel value of each channel is greater than the set threshold in the RGB space, and marks the pixels that exceed the threshold. Then the image is converted from RGB space to YCbCr space according to the mapping relationship for bright color separation. The correction method is divided into single and dual-channel saturation correction and three-channel saturation correction. For single-channel and dual-channel saturated pixels, the algorithm divides the saturated area according to the consistency of the hue, and corrects the saturated channel by copying the intensity change information of the reference channel. Corrected twice horizontally and vertically, and compared with the original channel value, the larger value is regarded as a reasonable value. For the three-channel saturated pixels, the correction algorithm adopted is to increase the pixel value from the periphery to the center according to the intensity gradient value of each channel of the pixel in the three-channel saturation area, that is, gradually increase each of the adjacent pixels after correction on the scan line. The intensity value of each channel is the local maximum value. After completing all corrections, perform dynamic range compression, and finally output the processed image.

2.METHOD

2.1Single and dual channel saturation correction

The value of saturated channel C in the saturated region ($C \in [235,255]$) generates distortion due to the limited dynamic range of the camera, and even all jumps up to 255. The intensity change of channel value representing the image details is lost. In RGB color space, R, G, and B color channels are strongly correlated. The algorithm considers that the intensity change in the reference channel can be used to reconstruct the lost information of the saturated channel with the same image position. However, it is not advisable to directly copy the intensity value of the reference channel, because different channels have different overall intensity, and direct copy will cause obvious color distortion of the image. Therefore, it is necessary to determine the baseline strength value of each saturated section reference channel, and then copy the strength difference value to the saturated channel value, so as to obtain the value after pixel correction. The calculation formula is as follows:

$$I_{C_{new}} = I_C(m,n) + I_{C_{ref}}(m,n) - \rho_{s_i} \qquad (1)$$

2.2Three-channel saturation correction

The full saturation correction algorithm used in this paper is to increase the pixel value from the periphery of the three-channel saturated area to the center according to the intensity gradient value of each channel of the pixel, that is, to gradually increase the intensity value of each channel of the adjacent pixels that have been corrected on the scan line to obtain the local maximum value.

$$C_{middle} = C_b + G \times N_q / 2 \qquad (2)$$

C_{middle} is the correction value of each channel of the middl e pixel in the fully saturated section, Cb is the non-fully saturated pixel at the boundary of the fully saturated re gion, G is the gradient value, and Nq is the number of fully saturated pixels in the fully saturated section.

The correction value of each channel of the remaining pixels in the full saturation section can be obtained by decreasing t he correction value of each channel of the pixel in the middl e of the full saturation section, that is, the local maximum va lue, which is calculated by the following formula:

$$C_{all} = C_{middle} - D \times G \quad (3)$$

Where C_{all} is the corrected intensity value of a certain channel of a certain fully saturated pixel, and D is the distance from the fully saturated pixel to the middle pixel of the fully saturated section. Finally, the correction result is trimmed to obtain the correction result of the fully saturated pixel.

Subsequent processing

When the single and dual-channel saturation correction and the three-channel saturation correction are all completed, the image pixel value must be greater than the 8-bit maximum value of 255, so it cannot be saved as an 8-bit RGB image. To solve this problem, the dynamic range of the image needs to be compressed. The compression formula for each channel is as follows:

$$R' = (R/M)*255$$
$$G' = (G/M)*255$$
$$B' = (B/M)*255 \quad (4)$$

Crrection algorithm

Algorithm design and possibility

$$X_s^* = Sth + p * b' * |X_k - Y_k| \quad (5)$$

In the formula

X_s^* is the corrected value of the pixel saturation channel;
Sth is the saturation threshold of each channel;
X_k is the value of the unsaturated channel of the pixel;
Y_k is the statistical value of the unsaturated channel at the saturated boundary of the saturated pixel and color region.
b is the proportional relationship, which is obtained by statistical calculation of the same color area.
p is the adjustment coefficient, in order to prevent the data from being too large or too small after correction, and to avoid algorithm failure.

1.The idea of Color Line algorithm considers the original image gamma correction to be approximately linear.
2.Assefa's algorithm idea is to directly superimpose the data of the channel with the least saturated pixels on the saturation threshold of the channel with more saturated pixels.
3.The algorithm in this paper directly corrects the model data in the figure without performing gamma correction. The algorithm in this paper is modeled as the following formula:

$$X_s^* = Sth + p * b' * |X_k - Y_k| \quad (6)$$
$$brb' = (R_l + R_r)/(B_l + B_r) \quad (7)$$

brb' is the ratio of the sum of the red channel values on both sides of the saturation boundary of the input image to the sum of the blue channel values.

When the image is read in the first pass, at the boundary from the unsaturated pixel to the saturated pixel, the on-chip RAM is used to store the 24-bit value of the three-channel value of the unsaturated pixel. That is, only the left side value is stored, and the right side value is obtained when the value is read backward. The corrected proportional relationship b'can be obtained by adding two sets of values on both sides of the saturation boundary and then performing operations.
In order to realize fast hardware correction under the premise of sequentially reading images, this paper adopts a hardware correction model with parameter b'.

Theoretical value after processing of each channel in RGB space

In terms of statistics, by processing the actual image and continuously fitting, the approximate relationship between the three channels can be obtained, and the saturation threshold of 235 is used for composition, and the saturation of each channel before processing is shown as follows

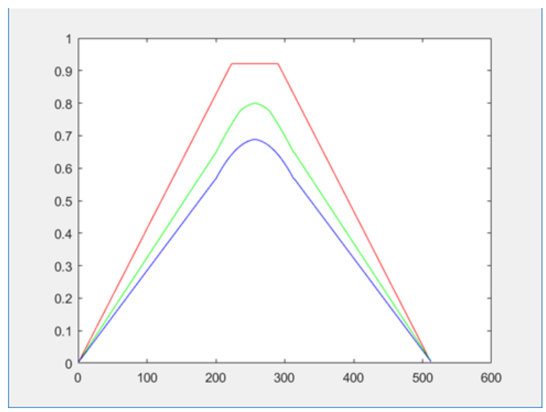


Fig.1 (a) channel value before processing

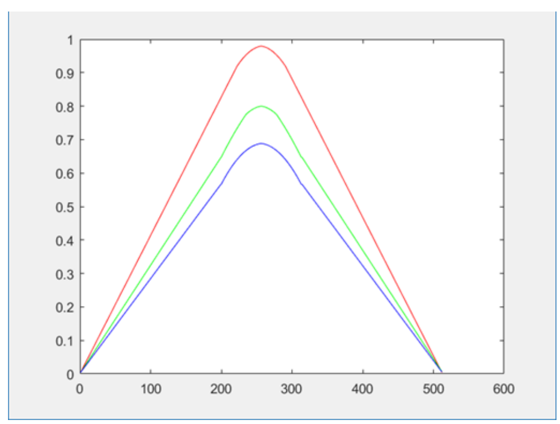


Fig.1 (b) channel value after processing

The model algorithm in this paper does not use gamma correction, which can restore the maximum value to between 250 and 251, eliminating the need for gamma correction, the effect is better, and it provides the possibility for hardware implementation.

3 Hardware implementation

3.1Preprocessing of original image information

The image information preprocessing module needs to extract the saturated channel data of the saturated pixel and the unsaturated channel data of the saturated pixel, as well as the data of the unsaturated pixel adjacent to the saturated pixel, and the position relationship with other pixels. It also needs to be based on the correction algorithm The color space that belongs to, performs spatial transformation on the image data, and prepares for saturated pixel correction.

In order to perform fast hardware correction, the ideal situation is to read a complete image in a sequence. In fact, this ideal situation is difficult to achieve due to hardware resources such as memory size.

3.2 Quantization after all saturation correction is completed

After saturation correction, the dynamic range of the image data has been expanded, which has exceeded the displayable color space and needs to be requantized to the displayable color space. Since the maximum value of the image data and its position in the image can only be obtained after the correction is completed, the quantization must be performed independently after the correction is completed. In order to perform fast hardware calibration, it is the key to this module to spend as much time as possible to read the complete image in one sequence to obtain the final data of all pixels.

3.3 Hardware resource calculation

3.3.1Clock requirements

The hardware implementation should consider the issue of processing time. For 24-frame video, it is related to the video resolution.

Common video resolutions are as follows:

80x320, 640x480 SD

1024x720p, 1280 x 1024p HD

1600x1280p, 1920x1080p Full HD

3840x2160, 7680x4320 Ultra (high) definition

It takes at least 3 times to process a frame of image, and the formula is as follows

$M*N*3*24 \le F$

M and N are the picture size, and F is the FPGA clock frequency.

For 1024x720p HD video, F should be greater than 54MHz.

For 1280x1024p HD video, F should be greater than 95MHz.

For 1600x1280p full HD video, F should be greater than 148MHz.

3.3.2 Memory requirements

For each pixel, at least 24 bits of RGB intensity value and 4 bits of saturation mark are used when the picture pixel is read into the memory in the first pass. For a 1024 x720p image, 20 bits of data are required to mark the pixel position, and 33.75MB is required for all storage. The second pass takes the same space to read in and correct. Assuming that there is 10% pixel saturation for processing and storage, the final synthesis of the two processing results also takes up storage space after dynamic range compression, so at least 108MB of storage space is required for 1024 x720p pictures , For 1280x1024p at least 196MB, for 1600x1280p at least 312.5MB. This method of storage is extremely storage space consuming and has high requirements for memory throughput.

This article uses the method to input by line as an example. The first line is entered for saturation determination mark, and then the second line is entered for marking while the first line of saturation correction is performed. After the correction is completed, the processing result is stored, and the occupied space is released after data processing. The released space can be used to store the data of the next row. Input by column is processed in the same way, which can greatly improve the memory utilization. For 1024x720p pictures, about 6.75MB of storage space is needed, for 1280x1024p pictures, about 12.25MB, and for 1600x1280p pictures, about 19.5MB.

3.3.3 Memory type selection

The memory of FPGA development board is divided into on-chip storage and off-chip storage.

After considering various characteristics and parameters, we decided to use SSRAM storage.

3.4 Hardware selection

On the basis of considering various parameters, we decided to use Altera DE4 high-end development platform, the components are as follows

3.4.1 FPGA configuration

JTAG and FPP configuration	Built-in USB download cable circuit

3.4.2 Memory

64 MB flash memory 16-bit data bus	2MB ZBT SSRAM	EEPROM for I2C communication
Two DDR2 SO-DIMM slots	400 MHz frequency clock	Maximum theoretical bandwidth 102 Gbps
Up to 8GB total capacity	SD card slot	Support SPI and SD 1-bit SD Card reading modes

3.4.3 On-board clock

Configure 3 programmable PLLs by FPGA	HSMA,HSMB transceiver clock source	SATA reference clock
FPGA LVDS clock input	50MHz/100MHz oscillator	

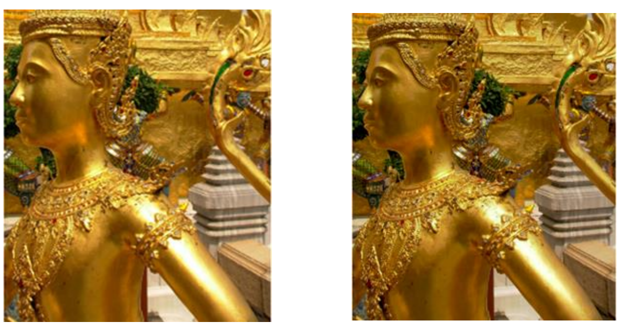
3.4.4 Power supply

DC input 12V and 3.3V	On-board power measurement circuit

3.5 Correction result

（a）Oversaturated image (b) Result of our method

FIG. 2-1Comparison of image correction results



(a)Oversaturated image (b) Result of our method

FIG. 2-2Comparison of image correction results

(a) Oversaturated image (b) Result of our method

FIG. 2-3Comparison of image correction results

4.CONCLUSION

The correction method used in this article is instant correction, that is, the pixels of the previous row are processed while scanning the pixels of the next row. The above results show that the new model hardware algorithm has greatly improved the image correction speed and resource occupation. The conclusion is that FPGA is feasible and efficient for high-speed image and video processing, and it can be done frame by frame for high-definition video. It has broad practical application prospects.

5.ACKNOWLEDGEMENT

The author would like to thank Assefa M, Poulie T, Kervec J, Shao Dan, Han Jiawei,their valuable information. In addition, we thank the reviewers for their helpful comments to improve the clarity of this article.

6.References

[1] Jun Fu, Wenbo Ji, Xuanqin Mou, "Correcting saturated pixels in images," Proc. SPIE 8299, Digital Photography VIII, 82990S (24 January 2012); doi:10.1117/12.908129

[2] Fu Jun, MOU Xuanqin, Ji Wenbo. Saturation image correction method for bright color

[3] Assefa M, Poulie T, Kervec J, et al. Correction of over-exposure using color channel correlations[C]// Signal and Information Processing. IEEE, 2014:1078-1082.

[4] Wu Yungang. Research and Design of RAW Image Saturation Correction Algorithm based on FPGA [D]. Xi 'an: Xi 'an Jiaotong University, 2016

[5] Wang Jing, Research on color Image Segmentation Algorithm based on color space segmentation, China Excellent Master Thesis Full-text Database, 2010,(01):-

[6] Shao Dan, Han Jiawei, Conversion between YUV and RGB, Journal of Changchun University 2004,(04):53-55. (in Chinese)

Author Biography

Jun Fu,he received the B.E. degree in Radio Technology at Xi'an Jiaotong University in 1997. He received the M.S. degree in Electronic Science and Technology at Xi'an Jiaotong University in 2000. He is currently working towards the Ph.D. degree in the Institute of Image Processing and Pattern Recognition, Xi'an Jiaotong University.

Work Experience:

1997-Present: Department of Information and Communication Engineering,

Xi'an Jiaotong University, Xi'an, Shaanxi, China.

2001-Present: National Data Broadcasting Engineering and Technology Research Center,

Xi'an, Shaanxi, PRC. China.