DESKTOP SUPERCOMPUTING TECHNOLOGY FOR SHADOW CORRECTION OF COLOR IMAGES

Artem Nikonorov, Sergey Bibikov
Samara State Aerospace University named after academician S. P. Korolyov, Moskovskoe shosse 34, Samara, Russia

Vladimir Fursov
Image Processing Systems Institute of Russian Academy of Science, Molodogvardeyskaya st. 151, Samara, Russia

Keywords: Image processing, Shadow distortion, Color correction, Identification, CUDA.

Abstract: The paper deals with information technology for correction of shadow artifacts on color digital images obtained by photographing of paintings with the purpose of their reproduction. Shadow artifacts are caused by differences of light intensity. The problem of shadow detection and subsequent color correction is solved. The architecture of heterogeneous CPU/GPU – system implementing the elaborated technology is considered, examples of real images processing are given.

1 INTRODUCTION

Digital images of paintings are used to create fine art reproductions, virtual museums, and digital catalogues in museums. Usually, they can be obtained in two ways: either by direct digital photographing, or by scanning negatives and slides made earlier. The main problem in picture photographing is to provide a uniform intense lighting of the entire picture surface. For this purpose, paintings are lighted from both sides by floodlight projectors. This causes the appearance of shadows along the edges of the frame.

Paper (Cheng, 2006) deals with the problem of removing the shadow using two images of the same object, one obtained in the "natural" lighting, and other made with flashlight. The problems of color reproduction and correction of color artefacts on digital images were also solved in papers (McCamy, 1976, Gevers, 1999, Weis, 2001, Fursov and Gavrilov, 2004, Nikonorov and Fursov, 2004).

In these papers, cases of uniform shadowing along the image area were investigated.

In this paper, we consider an information technology for removing the shadow in a situation, when the single unique copy of painting is available, and the shadows are located in known area. The paper deals with correction methods based on applying an algorithm for histogram transformation and identification of formal transformation model of lightness function. The architecture of heterogeneous CPU/GPU-system, implementing the developed technology is considered, examples of real images processing are given.

2 PROBLEM DEFINITION

Usually, the lighting equipment does not cause any perceptible color distortion. That is why in the present paper, to detect and remove the shadow distortion, we only use information about lightness of pixels. It is known (Judd and Wyszecki, 1975) that this information is contained in the lightness components of some color spaces. Specifically, the CIE Lab color space is used in the present work.

In the case when the lighting causes any additional color distortion, correction in color channels is performed. Due to the fact that the distortions are distributed uniformly across the image, the task can be solved after removing the shadows by methods considered in papers (Gevers, 1999, Weis, 2001, Fursov and Gavrilov, 2004, Nikonorov and Fursov, 2004).

Let us give the definition of a shadow. Let k be a number of row. Consider an interval $[0, N]$ in this

Nikonorov A., Bibikov S. and Fursov V. (2010).
DESKTOP SUPERCOMPUTING TECHNOLOGY FOR SHADOW CORRECTION OF COLOR IMAGES.
In *Proceedings of the International Conference on Signal Processing and Multimedia Applications*, pages 124-129
DOI: 10.5220/0002992101240129

row. Let $s^* < N$ be some point within the interval. Assume that for the k-th row of the uniformly illuminated image, the equality of mean lightness within the intervals to the left and right from this point is valid:

$$\frac{1}{s^*}\sum_{i=1}^{s^*} L_k\left(x_i\right) = \frac{1}{N-s^*}\sum_{i=s^*+1}^{N} L_k\left(x_i\right), \tag{1}$$

where $L_k\left(x_i\right)$ are values of lightness at the x_i-th points of the k-th image row.

As a shadow on the interval $\left[0, s^*\right]$ we shall call a distortion, whereby for any interval $\left[s^b, s^e\right] \in \left[0, s^*\right]$ holds the following inequality

$$\frac{1}{s^e - s^b}\left(\sum_{i=s^b+1}^{s^e} \widehat{L}_k\left(x_i\right) - \sum_{i=s^b+1}^{s^e} \breve{L}_k\left(x_i\right)\right) > d, \tag{2}$$

where d is a parameter characterizing the shadow intensity. Here and elsewhere, the values of lightness for illuminated pixels are designated as $\widehat{L}_k\left(x_i\right)$, and for shaded pixels as $\breve{L}_k\left(x_i\right)$.

Due to the light source is not a point source, the shadow boundary can be fuzzy. In this case, the definition of shadow (1) and (2) still holds, if the diffusion function is smooth and has no ridge point (sign reversals of the derivative). In this case, as a frontier point s^* of shadow field we take a point, where the diffusion function is equal to half-sum of its extreme values. From now on this point will be called middle point of diffusion (transition) area.

Furthermore, the shadow area for various rows k may differ because of the relief of picture frame. Thus, the shadow has two regions with different distortion, consequently, the methods for their correction must differ, and the boundaries of these regions are in general not rectilinear.

Area with uniform shading adjoins image edge. Between this area and illuminated area is a transition area, generally more narrow, which shading is decreasing in direction of the picture edge. The boundaries of the transition field are curved.

Thus, the problem is to determine the boundaries and to develop the algorithms for color correction in each area. The proposed technology aims to implementation in a distributed computing system.

3 DETECTING A BOUNDARY OF SHADOW AREA

The problem of detecting the boundaries of shadow transition area is problem of edge detection. The most approximate to it is the problem of edge detection in the CIE Lab color space considered in (Murashev, 2009). In this paper, active contours method (Kass and Witkin, 1987) was applied. But in our case, this method is not applicable in original form because the contours can be fuzzy. In the following, we consider the algorithm for detecting the boundaries of shadow transition area.

The problem is solved in two steps. First, we determine a set G of points $\left(x_k, y_k\right) \in G$ of transition area midline. Then, the entire transition area is formed in neighborhood of midline.

Let us consider a lightness component L of a three-component color image. To detect the boundary of shadow area let us introduce a criterion. For the point s^*, being a boundary of the shadow area determined by Eq. (1) and (2), the so called contrast indicator is introduced:

$$f\left(s\right) = C_R - C_L, \tag{3}$$

where

$$C_R = \frac{1}{N-s^*}\sum_{i=s^*+1}^{N} \breve{L}_k\left(x_i\right),$$
$$C_L = \frac{1}{s^*}\sum_{i=1}^{s^*} \widehat{L}_k\left(x_i\right)$$

are estimates of the mean value of lightness function within the intervals $\left[s^*+1, N\right]$ and $\left[1, s^*\right]$ respectively, and N is a number of lightness values of image row.

To show the applicability of the criterion (3) for detecting the shadow boundaries, together with the assumption (1), suppose that the variations of interval boundaries in the neighborhood of the boundary s^* for the case of uniform illumination do not cause changes of mean values.

$$\frac{1}{s^1}\sum_{i=1}^{s^1} \widehat{L}_k\left(x_i\right) = \frac{1}{N-s^2}\sum_{i=s^2+1}^{N} \widehat{L}_k\left(x_i\right), \tag{4}$$

$$\frac{1}{s^1}\sum_{i=1}^{s^2} \breve{L}_k\left(x_i\right) = \frac{1}{N-s^2}\sum_{i=s^2+1}^{N} \breve{L}_k\left(x_i\right). \tag{5}$$

Here, $s^1, s^2 : s^1 \le s^2$ are any points in some neighborhood s^*, including $s^1 = s^*$; $s^2 = s^*$ or $s^1 = s^2 = s^*$.

Thus, point s^* is the boundary of shadow area at the k-th row, if the contrast indicator (3) is distinct from zero and assume the maximum value at this point. Consequently, to detect the boundary of shadow area it is necessary to compute the values of the contrast indicator (3) at every point of the row, to choose point of maximal value, and to verify that the contrast at this point is perceptible enough.

We should emphasize once more that in reality the statements (1), (4), and (5) do not hold exactly. But the differences between the means within the intervals are generally much less than the contrast caused by shadow. That is why described procedure is effective in a wide range of shading intensity.

4 DETECTING THE BOUNDARIES OF A TRANSITION AREA

In the course of correction of the transition area the fact that the width of this area is varying along shadow boundary should be taken into account. This is caused by different distances between parts of the frame and the light-source. In particular, for the points of the frame more distant from the light source, the transition area is getting wider. In this section, procedure of detecting the boundaries of transition area with regard for the described peculiarities is considered.

The idea is to approximate the lightness function within the transition area by the steady increasing function. Actually lightness functions may essentially differ owing to shadowing of various picture elements. This is why approximation is implemented with respect to some averaged lightness function values obtained by summing the lightness of several neighboring rows.

To detect the boundaries of transition area within the certain local area, we have to solve a problem of approximating the set of averaged lightness values by following function

$$w(x) = \frac{c_2 (x - c_3)}{1 + c_1 |x - c_3|} + c_4 . \tag{6}$$

Estimates of the parameters $c_i, i = \overline{1,4}$ for each midline of local subset of rows are made applying the least absolute values method. As an initial approximation for subsequent rows, estimates of the parameters obtained in the previous step are used. As a result, we have a set of function (6) parameter estimates for each row.

Using the amended approximated lightness function of transition area for middle of each local subset of rows, the boundary values of transition area are defined by

$$x_{bor} = x_0^* \pm \frac{b(1 - \Delta_{bor})}{a \Delta_{bor}} , \tag{7}$$

where x_0^* is the amended coordinate of a midline point, $\Delta_{bor} = \Delta w(x) / w_{\max}(x)$, and $\Delta w(x)$ is admissible deviation of lightness function from the maximal value of approximating curve at the boundary point of transition area.

Moreover, for each row in the neighborhood of the amended midline point there is a set of values of the approximated lightness function which, in fact, are weighting factors characterizing a change of shading intensity.

5 ALGORITHMS FOR COLOR CORRECTION

Algorithms for color correction of shadow distortions in uniformly shaded and transition areas are developed in different ways. When solving a problem in the Lab color space only one component (lightness component L) is responsible for a shadow. Thus, for shadow correction it seems reasonable to apply equalization of histogram (Soifer at al., 2009) method for correction of grayscale images.

In this case, the histograms of lightness function should be built for uniformly shaded and illuminated areas adjoining to the transition area. Afterwards, the pixel-by-pixel transformation of lightness values in the shaded area is performed. This transformation converts the shadowed area histogram to a form of illuminated area histogram by means of closeness criterion minimization. The transformation is applied to all picture elements from the shaded field.

Application of this method to real images revealed the main defect - posterization effect, i.e. merging of different, but neighboring lightness values as a result of transformation. In this paper, we develop an approach proposed in (Bibikov and Fursov, 2008) and based on solving a problem of identification of color transformation model. Consider peculiarities of its applying for the present case.

Designate the coordinates of two points selected from the shaded and illuminated image areas as (x_i^l, y_i^l) and (x_i^r, y_i^r) respectively. By assumption, at uniform picture illumination, the colors of these points are similar:

$$L^*\left(x_i^l, y_j^l\right) \approx L\left(x_i^r, y_j^r\right). \quad (8)$$

In the following, the points meeting the condition (8) will be called matching points. Take into consideration the transformation function $\Psi[*]$ establishing a relation between the brightness values of matching points as follows:

$$L\left(x_i^l, y_i^l\right) = \Psi\left[L\left(x_i^r, y_i^r\right)\right].$$

Suppose $\Psi[*]$ is a polynomial, then coefficients can be obtained by identification using least squares method or least absolute values method.

To take into account shadowing decreasing along rows a set of weighting factors obtained by approximation (7) of lightness function within the local subset of rows is used. The transformation of lightness values in each row of transition area is performed according to the next rule:

$$L_i^*\left(x_i\right) = L_k\left(x_i\right) + w_{ki} \cdot \tilde{\Delta}_L,$$
$$\tilde{\Delta}_L = \Psi\left[L_k\left(x_I\right)\right] - L_k\left(x_I\right),$$
$$w_{ki} = \frac{w_k\left(x_i\right) - \min w_k}{\max w_k - \min w_k},$$

where $\Psi\left[L_k\left(x_i\right)\right]$ is a transformation for the point x_i of the k-th row, w_{ki} is weighting factor of each point in k-th row, and x_I are coordinates of the nearest shaded point of row from the uniformly shaded area.

6 IMPLEMENTATION OF THE DESKTOP SUPERCOMPUTING SYSTEM FOR IMAGE CORRECTION

Developed software package is designed for solving the full range of tasks related to computer-aided color correction of images including point artefacts removal (Fursov and Nikonorov, 2008). Software implementation provides for operation both within existing image processing systems such as Adobe Photoshop and in free running mode.

Some of the developed algorithms, in particular, the algorithm for identification of point flares, are compute-intensive. As shown in Fursov and Nikonorov, 2008), implementation of computations using Nvidia universal graphics processor and CUDA technology allows the acceleration up to 10 times to be obtained. These investigations show that utilizing GPU allows the large-scale images to be processed in real time, for instance, in a video stream.

With regard to what has been said, the software should meet the following requirements. First, the software is to operate both in the free running mode and in the integration mode with existing image processing systems as well. Second, the most resource-intensive computations should be carried out in parallel, using GPU. Besides, the software should meet the requirements of cross-platformness. The architecture of software package meeting the said requirements is represented in Fig. 1.

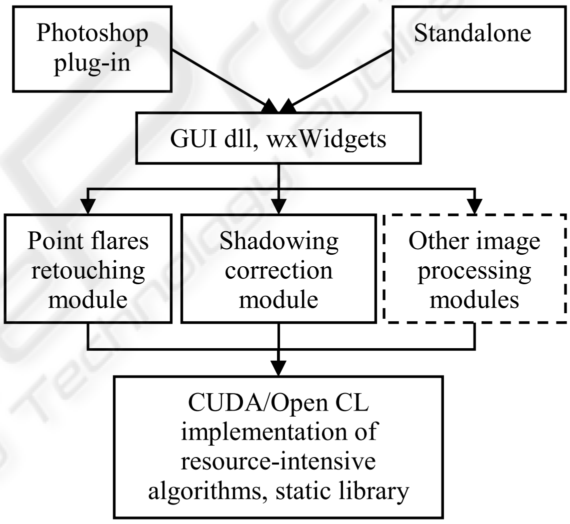

Figure 1: Architecture of software package.

As shown on the figure, the system structure may be divided into several levels. First level is the user's interface. Midlevel is the implementation of graphics interface (GUI) of the computer-aided image retouching. Third level that is independent from other levels is the dynamic libraries, where computational algorithms are realized. If used algorithm is compute-intensive and the CUDA supporting hardware is available, then the algorithms can activate procedures implemented for GPU. In the future, it is intended to make the GPU computations more universal by using OpenCL (Munshi, 2009). Such approach will allow both the GPU Nvidia and the GPU AMD (ATI) to be utilized for computations.

If the GPU is not available, the computational procedures may be carried out by several CPU

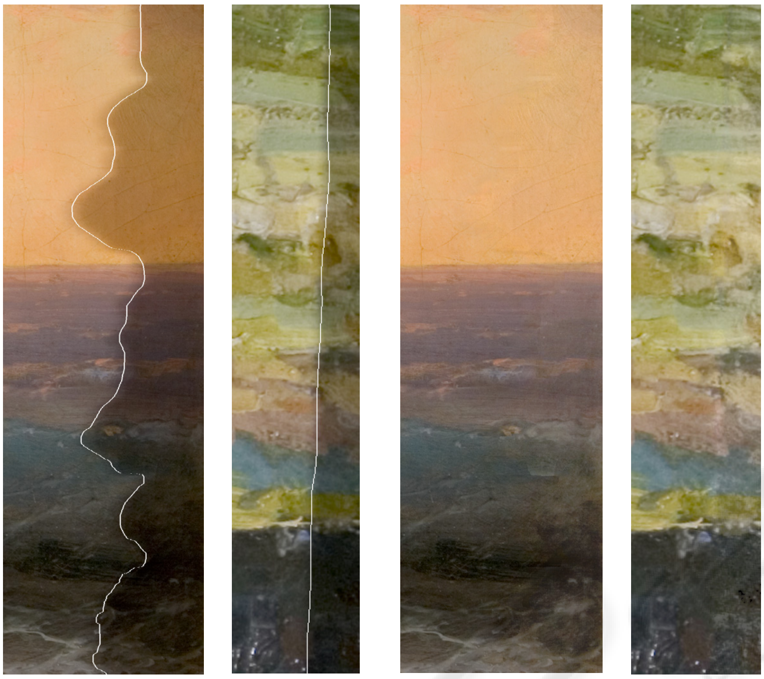

Figure 3: Example of color correction of images with the curvilinear boundary of shadow stripe and area of slowly changing color: a), c) initial images with the detected shadow line; b) d) images after correction.

threads. The multithreaded implementation is performed using OpenMP. Most of the developed retouching algorithms allow data decomposition to be performed, which allows parallelizing to be carried out in quite a simple way.

Though the system GUI is connected with the algorithmic part by dynamic linking, in this case, the cross-platform C++ library is used to create the GUI. At the present time, several libraries of this kind are known. The most popular of them are QT, GTK and wxWidgets library (Smart, 2000). In our opinion, wxWidgets is distinguished for its conceptual integrity and flexibility in using, which fully justifies its utilization in the present work.

Developed system intended for processing images interactively and in our implementation CUDA shadow correction algorithm can process one image at a time. Image divided into small fragments consist of some rows of original image and each of them processed by one CUDA kernel. Most of the developed retouching algorithms allow the data decomposition to be performed (Nikonorov and Fursov, 2008), which allows parallelizing to be carried out in quite a simple way.

7 RESULTS OF EXPERIMENTS

As test images, high-resolution digital photos of real paintings featuring a grove and sunset were used. The images processed have the size 554×3558 and 145×3933 pixels respectively. Image format is TIFF, color space is Lab (8 bit per channel). Enhanced fragments of two initial images with shadow area are shown in Figure 2a and 2b.

The image featuring a sunset is more complex in respect of detecting the shadow boundary. Main difficulty consists in large dimensions of the transition area (up to 120 pixels). Besides, the width of this field is varying within the wide limits along midline.

On the image featuring a grove, the shadow boundary is rather distinct; this is determined by small width of transition area, whereby the boundary is almost rectilinear. Both images feature a large number of contrast elements and are characterized by a wide range of lightness variation in channel L. Moreover, both images are noisy. Said peculiarities essentially complicate identifying the weight function (6) in the transition areas.

Mean values within the shadow transition area of images featuring a sunset and grove are represented by a white line in Figure 3a and 3b. End result after shadow artifact removing is represented in Figure 3c and 3d.

In this paper we present only two shadowed images, but our system was tested by prepress specialists on about 200 images and sufficient quality was obtained in 75% cases. In other cases additional correction was required. Automation of shadow correction process reduces time of processing one shadowed image from about one hour to 2-3 minutes.

8 CONCLUSIONS

The developed information technology for color correction of shadowed images admits a high automation degree. Its software implementation has resulted in essential reducing the time expenditures for prepress of colorful images. Apart from preparation of painting reproductions, the technology may be employed to process the areas of images obtained by aerospace monitoring systems and intended for print, and to provide the services of improving the quality of digital images to a wide range of users.

There is a difference in internal parallelism and computational complexity degree between the steps of the technology developed. Accounting for these factors when elaborating a program complex, in particular, implementation of image processing algorithms in the GPU, allows its productivity to be increased.

ACKNOWLEDGEMENTS

We render our thanks to the specialists of Agni Publishing House for their qualified assistance in computer-aided color correction and testing of software on real images.

This work was supported by the Russian Foundation for Basic Research (Project No. 09-07-00269-a).

REFERENCES

Cheng L. Removing shadows from colorful images, *PhD Thesis* – Simon Fraser University, 2006. – 155 p.

Weiss Y. Deriving intrinsic images from image sequences, *ICCV01* – IEEE, 2001. – V.II – P. 68-75.

McCamy M., Davidson J. G. A color-rendition chart, *J. App. Photog. In Eng.* – 1976 – V.2 – P. 95-99.

Gevers T., Smeulders A. W. M. Color-based object recognition / *Patt.Rec.* – 1999. – V.32 P. 453-464.

Fursov V. A., Gavrilov A. V. Parallel algorithm of data selection using relative conforming estimate criterion / *Proceedings of The 12th ISPE International Conference on Concurrent Engineering: Research and Applications*, Ft. Worth/Dallas, U.S.A., 25 - 29 July, 2005, p. 375-380.

Fursov V. A., Nikonorov A. V. Conformity estimation in color lookup tables preprocessing problem. *7th International Conference on Pattern Recognition and Image Analysis: New Information Technologies, St.Peterburg*, 2004, Russian Federation, Volume I, p. 213-216.

Nikonorov A. V. Constructing the conforming estimates of non linear parameters, *4th European Congress on Computational Methods in Applied Sciences and Engineering*, 2004, Finland. – PP. 404, 429.

Judd D., Wyszecki G., *Color in business, science, and industry* – New York: Wiley, 1975. – 553 c.

Murashev D. M. Automated cytological specimen image segmentation technique based on the active contour model, *Proceedings of Moscow Institute of Physics and Technology*. – 2009. – V.1 – N.1. P. 80-89. – (in Russian).

Kass M., Witkin A., Terzopoulos D., Snakes: Active contour models, *Int. J. Computer vision.* – N.1. – 1987. – PP. 321-331.

Soifer V. A. et al. *Computer Image Processing* / – Lightning Source Inc, 2009 – 296 pp.

Bibikov S. A., Fursov V. A., Color correction based on models identification using test image patches, Computer optics. – Samara-Moscow, 2008 – T.32 № 3. – C 302-307 – (in Russian).

Smart J. *Cross-platform GUI Programming with wxWidgets* – Prentice Hall, 2000. -714 p.

Munshi A. Kronos OpenCL Working Group, *The OpenCL Specification*. – 2009. – 314p.

Nikonorov A. V., Fursov V. A. Distributed computational system of color images correction, *XV All-Russian conference Telematica*. – 2008. – Vol.1. – p. 88-89 – (in Russian).