Detection and Correction of Errors on Digital Photos Due to Technical Imperfections in the Registration System

Jeroen HUYLEBROECK
Luc De BACKER
Lieven De STRYCKER
Jean-Pierre GOEMAERE

KaHo St Lieven, department of industrial engineering/electronics
Gebroeders De Smetstraat 1, 9000 Gent, Belgium
luc.debacker@kahosl.be

Abstract—This article describes the study of the detection and correction of diverse errors that can occur while taking digital photographs of flat objects such as pages from a book or magazine. This paper deals with uneven lighting conditions and the consequences thereof on the brightness of the photographed subject, geometric image distortions due to the technical characteristics and the chosen settings on the camera, and possible distortions because of the surface on which the photographed material is positioned. The processing of the data is done with Matlab. The user is supposed to make correct use of adequate reference material as a means for the identification of the imperfections in the registration system to make the necessary corrections. The final purpose of all these procedures is to reproduce the photographed material as truthfully as possible.

Keywords— image processing, Matlab, photography, reproduction

I. INTRODUCTION

With the introduction of digital technology in the photographic industry, new possibilities arose in the domain of reproduction, pre- and post-processing, synchronisation etc. The focus of this paper is the correct reproduction of flat objects photographed under non-ideal circumstances. When taking pictures, both analogue and digital, we are faced with a number of difficulties when it comes to reproducing an object (e.g. a sheet of paper with text and/or images) as exactly as possible. Reasons for this are to be found in uneven lighting conditions, shadowing, lens distortions and the positioning of the objects themselves.

In the past, when digital processing was nonexistent, most lenses were aimed at replicating the geometric qualities of the photographed object, while making some concessions on sharpness, brightness, etc. With the recent advancements in technology it is possible to shift this balance in the opposite direction. Minor problems in the lighting and geometric conditions can be solved in a satisfactory manner by the post-processing of the raw images shot by the camera. For that purpose, we will analyse the digitised photographs in Matlab.

For each aspect, a specific reference image is used. For the correction of the imperfect lighting situation, this reference image consists of a plain white paper, photographed under the same lighting and camera conditions as will be used afterwards for the final photograph. Thanks to this reference image we can compensate for the lighting inadequacies. For the correction of geometric distortions we use another piece of paper with a number of evenly spaced black dots. Using these dots, it becomes possible to make an estimation of the geometric faults that are introduced by the imperfections of the lens, the positioning of the object etc. The necessary corrections can be determined and afterwards applied to the uncorrected image thus obtaining the final rendering of the object that has to be reproduced.

II. DETAILED EXPLANATION OF THE PROCEDURE

Before the actual processing stage, the pictures that are to be processed need to be verified. The criteria that have to be met are the following: stability of lighting conditions throughout the whole registration process of the reference material and the uncorrected image itself; correct; straight-up and identical placing of the photographed material, fixed white balance, identical shutter speed and exposure time; no environmental objects in any of the pictures, a light grey photographing surface to prevent the detection algorithm to interpret the surface as a "dot" or as the whitest point of the photograph.

In this paper we purposely use only one lamp to light the different materials. The photographing process is depicted in

Figure **1**. This results in a relatively strong lighting gradient on the photographs as is shown in

Figure **2**, a photo of a sheet of white paper, photographed under the circumstances as shown in

Figure **1**. This is the reference that is going to be used to determine the necessary lighting correction. Under normal conditions more lamps would be used, resulting in a less pronounced shading of the image.

When the criteria are met, the photographs, stored in a Matlab compatible format, can be loaded into Matlab. When loaded, these images are stored in a 3-dimensional

matrix of unsigned 8 bit integers, representing the RGB-values of each pixel (RGB stands for Red-Green-Blue).

Figure 1: Lighting procedure

Figure 2: Lighting reference

The valid range of each of these values is 0-255. To simplify and speed up the processing, the RGB colour space is converted to YCbCr colour space. The signification of the three colour space components Y, Cb and Cr are respectively the luma component and a chroma colour-difference component for blue and red. These are also represented by 8 bit values, but the valid values for the Y component vary from 16 for black to 235 for white. Cb and Cr are scaled to a different range of 16-240. This offers the possibility to only use the first plane of the 3D matrix for the intensity measurement.

To determine the brightest value, it is sufficient to search the intensity plane of the YCbCr matrix for the highest value. To eliminate pixel errors, the analysis can be preceded by a local median or mean interpretation of this plane. It is up to the user to determine whether or not this is necessary and what (square) area should be evaluated. The blank sheet reference is then processed. A new 2D offset matrix is created containing per pixel luma difference values compared to the value found earlier. This offset matrix is a valid lighting correction for all pictures taken under the same circumstances as the lighting reference (

Figure **2**). It is sufficient to store this matrix in memory, after which it is added to the first (Y-) plane of an uncorrected image of a white sheet containing text, figures, photographs, etc taken under the same circumstances. A simplified representation of the described workflow can be found in the flowchart depicted in .Figure 3. The next processing phase is the correction of geometric

deformations as shown in the rightmost branch of the flowchart in

Figure **3**. To evaluate the type and amount of geometric error, we need to use a second reference.
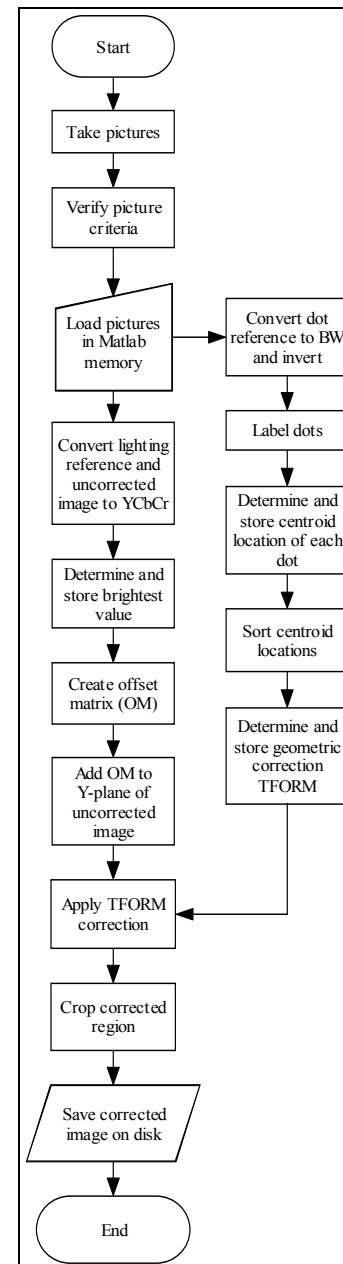

Figure 3: Flowchart

. This reference consists of a blank sheet of paper filled with a number of evenly spaced solid black dots. This reference is of course also photographed in the same way as the other material. An example is shown in

Figure **4**.

This step also demands another conversion and simplification to maximise the processing speed and reduce complexity to a minimum. One photographed dot consists of an amount of adjacent black pixels. We need to reduce this to a single point for the remainder of the process.

In order to achieve this, a conversion from the RGB reference to black and white is done after which the colours are inverted. The following step in this process is

labelling. Every group of pixels forming a dot is labelled accordingly, thus forming several separated groups.

Afterwards, all these groups can be processed at once and evaluated according to each group's centroid (centre of mass).

The location of each centroid is stored in a matrix, resulting in a representation of the geometric distortion by means of several points.
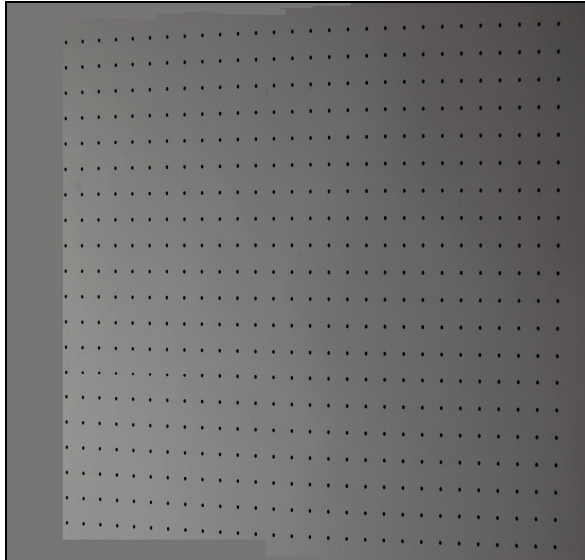

Figure 4: Geometry reference

Due to small inaccuracies caused by a slight rotation of the reference image and the chosen threshold. However, the dots are not always labelled in exactly the same order as they logically appear on the paper. This is why we have to re-sort the dot centres.

As we are working in quasi rotation-less conditions the Matlab labelling is already correct on a column basis and we only need to sort the dots in each of the columns. Once sorted, the dot reference centroid matrix can be compared to a distortion-less representation.
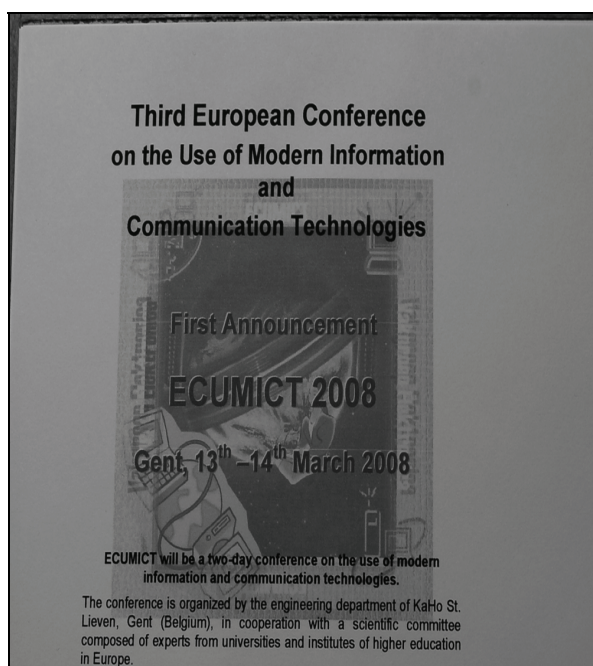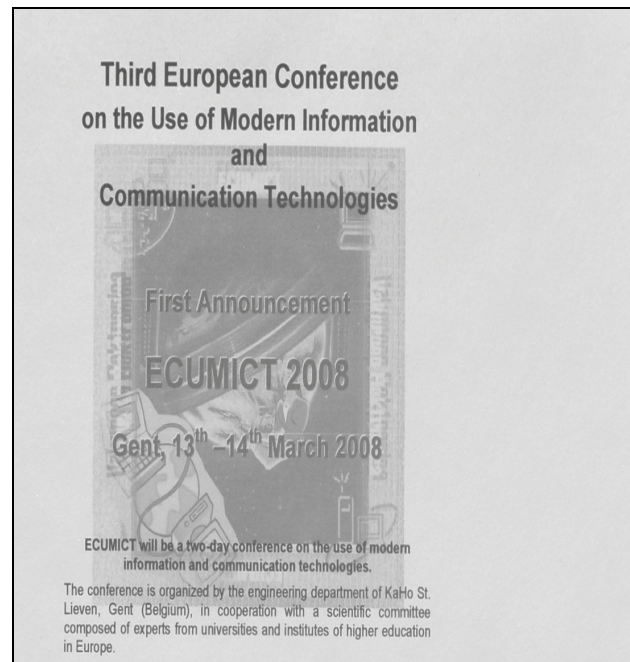

Figure 5: Original Image

Figure 6: Corrected image, cropped.

The reconstruction of the distortion-less dot reference centroid matrix is based on the measurement of the distances between the 4 centremost dots.

The mean value of these 4 dots is calculated to serve as the correct spacing value. This value is now used to construct the corrected matrix containing the location of each dot, evenly spaced with the aforementioned spacing value. When this is finished the uncorrected matrix is compared to the corrected one.

We evaluate on a stepwise linear basis between every dot and construct a Matlab TFORM (which stands for transformation) out of this.

This TFORM is then applied to the original image thus correcting the region determined by the presence of dots, which is then cropped to the corrected region of the image, thus resulting in an undistorted, final version of the taken pictures.

III.　　RESULTS

By working in YCbCr colour space the whole process of correcting the photographed object can be sped up by a measurable amount all the while preserving the overall tone of the image.

To guarantee the quality of the correction, the user has to make sure his photographic material meets the aforementioned criteria. The results of the lighting and geometric correction are available by means of an example in the form of figure 5, the original image, and figure 6, the figure corrected as mentioned in the previous paragraphs.

Both have been rotated 90° for clearer visualisation of the effects. Furthermore Figures 2-4 and 5 still have a part of the photographing surface in the picture, but this is automatically removed in the cropping process.

IV. FURTHER STUDY

The nature of the lighting can cause the contrast level of the picture to be compressed, as pure blacks and whites can be hard to reproduce both in digital and printed form. Automating a contrast and/or brightness adaptation is beyond the scope of this article as this is for the most part a process where human preference and hands-on interaction are wishful, which can easily be done in a later post-processing phase using Adobe Photoshop, Corel Paint Shop Pro, WSU's free Paint.net, or some other image processing program. Further study can also be conducted in the domain of colour/tonality correction and adaptation.

REFERENCES

[1] Degeest, Wim *Digitale Beeldverwerking*
[2] Davis, Timothy A. & Sigmon (2005), Kermit Matlab Primer Seventh Edition, CRC Press
[3] Ojanen; Harri (1999) *Automatic correction of lens distortion by using digital image processing*