Automatic Geometry Correction and Enhancement

Digital Image Processing (DIP) Final Project Proposal Report

王卓扬

I Introduction

I. 1 Concept

Geometry correction is the process of enhancing images by applying geometric transformations to known regions of the image with known actual shapes to correct them to the correct orientation (Figure 1.1).

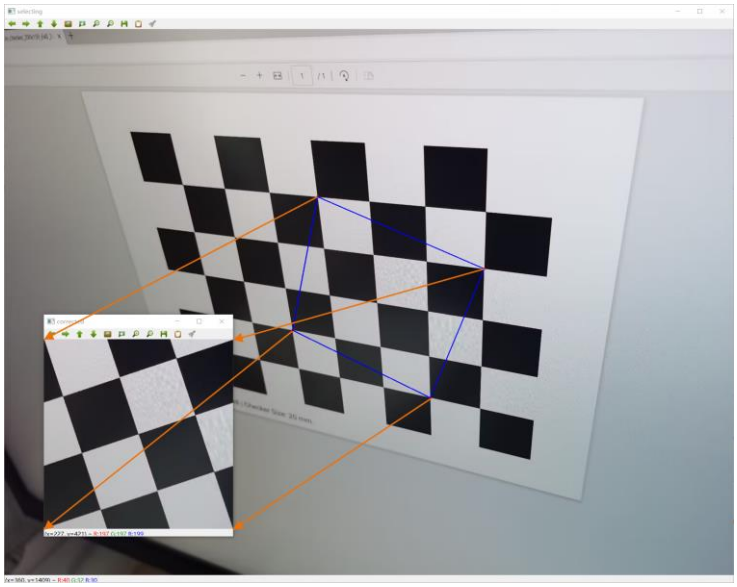

Figure 1.1 Geometry Correction

Automatic geometry correction means using image processing techniques to automatically search for border information in the image, extract the most likely region that needs to be corrected (such as the edge of a piece of paper), and perform geometric correction.

Enhancement refers to a post-processing operation on the image after geometric correction, which aims to make the image information clearer using image processing techniques.

I. 2 Significance and Application

Firstly, and most commonly, this technology can provide users with convenient document scanning support with just a high-resolution camera (such as a smartphone), without the need for scanning equipment or other hardware devices.

In addition, the programmatic image processing workflow can also provide users with batch image processing services.

The related technology has been implemented in some apps and tools, but the results are not particularly impressive, and there are almost no open-source projects available. Additionally, it heavily relies on commercialization and the algorithms lack significant improvements. There are also other issues, which will be discussed in the next section.

There are also other derivative applications besides the aforementioned. Scanned documents can be processed by OCR programs for text recognition. The automatic edge-cutting technology

ensures the adaptability of the algorithm in different scenarios, which is extremely helpful for tasks such as historical document organization. The process of extracting the plane where the document is located is equivalent to analyzing the three-dimensional information of the scene in a photo, which can be used as the basis for developing AR and other applications.

II Similar Work and Improvement Analysis

II. 1 Camscanner

There are some popular APPs have been focused on this field very early, such as *Camscanner* (Figure 2.1).

Figure 2.1 Screenshot from the official website of Camscanner

Camscanner provides a mobile scanning service that can perform geometric correction on documents in photos. The correction area can be selected by the user, and the program also provides suggestions automatically. In addition, it offers a range of supporting services such as image editing, enhancement, PDF export, cloud storage, etc.

II. 2 Shortcomings

However, Camscanner also has some shortcomings that need improvement:

1. In terms of user interaction design, compared to about a decade ago, *Camscanner*'s user interface design has become fancier, sometimes affecting the operation.

2. *Camscanner*'s automatic cropping function often fails to crop to the desired edge and its accuracy is not high, possibly due to different optimization directions.

3. *Camscanner* assumes that the user takes photos at a good angle, which may **lead to some issues with a more acute angle**. For example:

   a) If the **focus** is different for different areas at different distances, the software may not process the blurriness in different areas of the document.

   b) The actual shape of the object may **not match the shape** after correction, resulting in, for example, a rectangular shape instead of a square after shooting and correction of a square area.

   c) There are other related issues that are also the focus of my project to address.

4. *Camscanner* doesn't seem to have considered curved surface correction functionality, but in real life, we often encounter situations where the required information is on a curved surface, such as scanning one page of a thick book that is spread out.

5. Camscanner has become increasingly commercialized, which affects user experience and

blocks some practical features. Considering the non-profit perspective, developing a set of image scanning toolchains for personal convenience is meaningful in daily life.

6. There are other areas that can be improved, but we won't go into them here.

III Goals and Specifications

The project has a fixed main direction, but there are many specific functions that can be developed internally. Therefore, the main focus is on creating a complete, seamless, and user-friendly toolchain, rather than improving a single algorithm's performance on a single metric.

Based on this philosophy, I have divided the project goals into several stages. Apart from the basic functionalities, the other features will be completed and polished based on time and effort:

1. Reproduce the technique, including automatic edge slicing, geometry correction and common image enhancement algorithms. *(Fundamental)*

2. A better edge slicing algorithm. *Better* means faster, or higher accuracy, etc.

3. Scale distortion. Guarantee the shape of the corrected image is consistent with the origin object in the real world.

4. Vectorization. Or other things to improve the program performance.

5. Camera calibration. Take the distortion and the intrinsic parameters into the consideration. In addition, photometric calibration can also be considered to remove the effects of the photometric distortion.

6. Curved surface correction. We can model the surface or slice the curved surface into several planar surfaces to apply the algorithm into related situations.

7. Planar AR. Omitted.

8. Optical character recognition. Recognize the words on the document which is corrected and enhanced.

9. Transplant the algorithms into a mobile APP. That is what we called “Implementation”.

9. Some other things.

IV Methodology

In this report, we do not examine the implementation methods of other requirements - or other requirements have been briefly mentioned above. We mainly discuss the methodological ideas used in the two most critical parts of the entire system, namely, the automatic edge-cutting and geometric correction.

In traditional methods, automatic edge slicing generally consists of several steps, including edge detection, line segment extraction, and rectangle matching and comparison.

We generally use *Canny* algorithm to detect the edges, and then use some methods to pick out the line segments in the edge image, such as *Hough Transformation* or others. Then we can combine the highly evaluated line segments to form a series of rectangle areas. Finally we can pick out the area we want, or just show all of them and let the user to pick the best one.

In this rectangle evaluation step, we can also use the posterior knowledge in geometric correction to iteratively estimate whether the selected area is close to the true perspective projection shape of a rectangle.

In the geometric correction part, the method is very traditional, and we use homography matrix

to implement this step. Homography matrix represents the direct linear transformation relationship between two regions in space, which can be obtained from the correspondence between four pairs of points on two planes, and the four points we selected for the region selection are just enough. Once we obtain the homography matrix, we can perform a linear transformation on each point of the image to obtain the corrected image.

The scale distortion issue mentioned earlier is also involved, as we cannot directly obtain the actual aspect ratio of the selected rectangular region in the real space corresponding to the image. This can be solved by using the column vector equation of the parallel condition and solving the matrix null space to obtain the depth information of each point in the scale sense, thus calculating the aspect ratio. We will not go into detail here.

V Challenges

This project is not considered cutting-edge research, but rather a collection of engineering problems, so the difficulties encountered can always be resolved. Therefore, this part will be omitted.