TWO SIMPLE ALGORITHMS FOR DOCUMENT IMAGE PREPROCESSING

Making a document scanning application more user-friendly

Aleš Jaklič, Blaž Vrabec

Computer Vision Laboratory,Faculty of Computer and Information Science, University of Ljubljana, Tržaška 25, Ljubljana, Slovenija

Keywords: document scanning, Hough transform, detection, position, orientation

Abstract: Automatic Document scanning is a useful part of an information system at personal identification checkpoints such as airports, border crossings, banks etc. Current applications usually require a great deal of carefulness of the scanner operators – the document has to be positioned horizontally and special care must be taken to detect corrupt scans that can occur. In this work we describe ideas for two independent algorithms for the document rotation correction and automatic detection of corrupt scans. One algorithm relies on the Hough transformation and the other on brightness gradient of the image. The output of each algorithm is a cropped image of the document in horizontal orientation, which can be used as input for further processing (such as OCR). Also the estimate of scan corruption is returned. Also shown are some testing results of the algorithm prototypes written in MATLAB environment.

1 INTRODUCTION

Computer aided processing of personal identification documents is becoming increasingly important in places such as border crossings, banks, etc. Documents with integrated memory chips have already been developed but they are not common yet. That is why processing of images acquired with document scanners is more useful in document processing applications.

Numerous problems have to be solved in such applications. One of the first problems is detection of corrupt scans resulting from carelessness of the scanner operator. Another problem is detection of size, location and orientation (rotation) of the document in the acquired image.

In this work two independent algorithms are proposed for solving the problems mentioned. The output of the algorithms is a cropped image of the document in fixed orientation. Such images can be used for further processing (OCR for example).

The application that motivated this work is management of client entrance in a casino. Local laws in Slovenia require that every person entering a casino must identify himself with a personal identification document and that the entrance must be logged somehow. This is done much faster with the use of document scanners and computer processing as opposed to manual data entry.

2 THE PROBLEM DOMAIN

Figure 1: An example of a good document scan

All document processing starts with an image acquired using an optical document scanner. The image contains a personal document, which can be in any position and orientation, and also other distracting objects such as the operator's hand

Jaklič A. and Vrabec B. (2005).
TWO SIMPLE ALGORITHMS FOR DOCUMENT IMAGE PREPROCESSING - Making a document scanning application more user-friendly.
In *Proceedings of the Seventh International Conference on Enterprise Information Systems*, pages 116-121
DOI: 10.5220/0002554501160121

(Figure 1). In some situations the image of a personal document can be corrupt due to the carelessness of the operator and other movements during the scanning process (Figure 2). This can happen despite the fact that dedicated document scanners have very short scanning time – under one second.

Figure 2: An example of corrupt scan

In the area of document inspection, many things can be done with the use of document scanning. Personal data can be automatically read using OCR, the program can inspect the document security features, personal photograph can be extracted etc. The algorithms for these tasks may work better and faster under some general prerequisites – that the document be in horizontal orientation and cropped from the unnecessary background, and in some cases such prerequisites may even be required.

The algorithms described in this work focus on these general prerequisites of document scanning. The result of the algorithm should be a cropped image of the personal document in horizontal position suitable for further processing. An important problem is also the detection of corrupt scans. With this detection we can alert the operator that another scan should be made and avoid additional processing with useless results.

3 ALGORITHMS DESCRIBED

The sample image (Figure 1) shows that a good scan consists of a very bright rectangular shape (the personal document) and a much darker background. The idea of our algorithms is based on this observation. If we find the strongest edges that form a rectangle in the image, it is very likely that these are the edges of the document.

Corrupt document scans can be simply described as images in which the strongest edges do not form a rectangle, and we can estimate the corruption of the scan using this information.

Two independent algorithms have been developed for the problems mentioned. One is using the Hough transformation (Hough, 1962, Duda and Hart, 1972), which is a well-known method for edge parameterization. The other relies on the computed brightness gradient of the image.

3.1 Algorithm with the use of Hough transformation

The scanned image is first downsampled to the size that allows fast processing and is large enough for accurate pinpointing of the personal document borders. The simplest and fastest algorithm – nearest neighbor – is sufficient for this operation.

The next step is edge enhancement with the use of the Sobel filter that estimates the brightness gradient at each image pixel. We only need the size of the gradient and we discard the direction data. Other edge finding algorithms could be used, but the Sobel filter proved to be very satisfactory and it incorporates enough smoothing to remove the need of prior smoothing of the image.

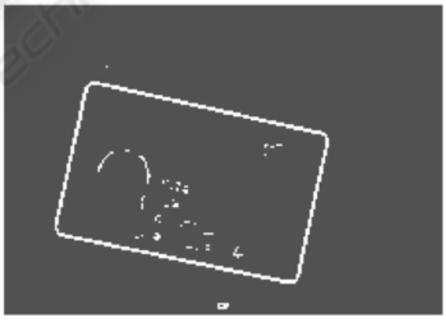

Figure 3: An example of thresholded image of gradient size (compare to figure 1)

The image of the size of the gradient is thresholded and the resulting binary image of edges (Figure 3) can be transformed with Hough transformation. In our case the borders of the document are represented as the local maxima in the transform image (Figure 4).

$$p(\varphi) = \sum_{d;H(\varphi,d)>m} H(\varphi,d) \qquad (1)$$

If we take a closer look at vertical projection p of the Hough transform H (to the φ axis) we can see that the ordinary projection results in a constant function. To make the projection useful we have to project only values over a certain threshold m.

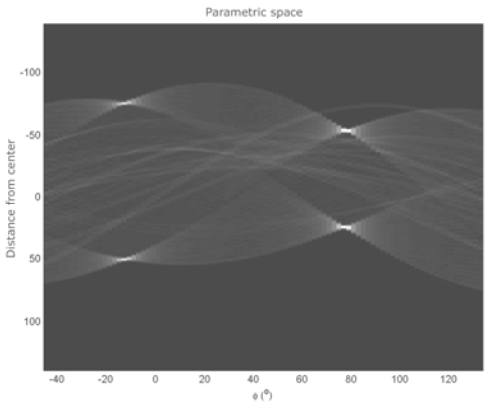

Figure 4: An example of Hough transform of an image of edges

Such altered projection normally has two notable local maxima – one at the angle of the vertical borders and the other at the angle of the horizontal borders of the document (Figure 5).

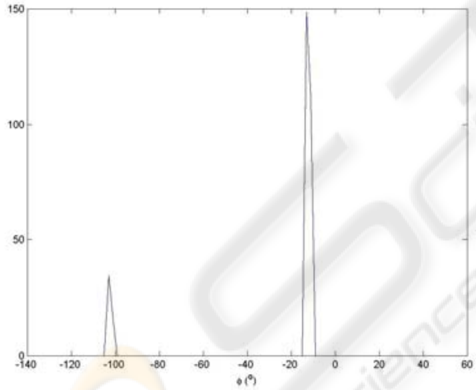

Figure 5: Vertical projection of the Hough transform

It is important to note that the two maxima are 90° apart and that the larger maximum corresponds to horizontal edges, which are longer in the case of an ordinary landscape oriented personal document. Of course, this is not quite true in the case of corrupt scans, where we can see more than two maxima or the maximum does not stand out so obviously (Figure 6).

The orientation (rotation) of the personal document in the scanned image can be calculated from these maxima. We can compare the vertical projection with a predetermined function that has appropriate maxima at 0° and 90° using the cross-correlation.

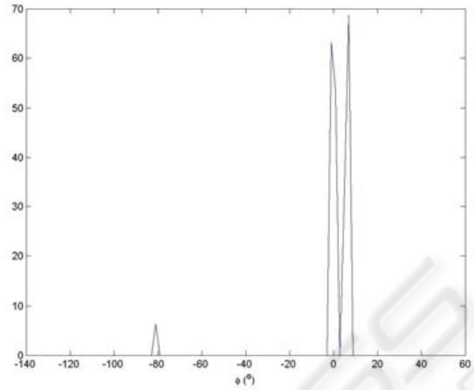

Figure 6: Vertical projection of a corrupt scan (compare to figure 2)

The lack of a strong maximum can be used for corrupt scan detection. In this case the maxima are more “widely spread” and not as “sharp”. We can estimate this feature by calculating the standard deviation around each of the two maxima (in the neighborhood ±45° of the edge) and we return the worse result of the two.

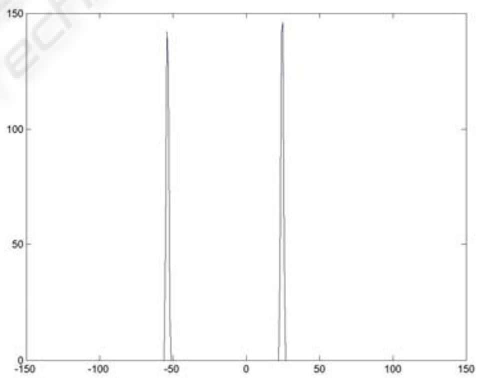

Figure 7: Vertical section of the Hough transform at the angle of an edge

Now that we have the document rotation we can look at the vertical sections of the Hough transform at the angles of the vertical and horizontal borders of the document (Figure 7). Normally, two local maxima of approximately the same size can be found in one section – the position of the two parallel borders of the document.

So now we have determined the rotation of the document and obtained parametric description of the four borders. We can derotate the scanned image and crop it at the borders (Figure 8). The localization of the borders is not exact because we have decreased the image size prior to computations and this introduces some error. Also the Hough transform is calculated at discrete angles, and thus some error is expected in the calculation of the document rotation.

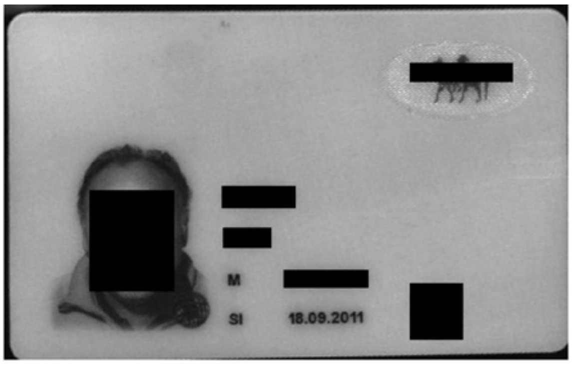

Figure 8: The result of the algorithm

3.2 Algorithm with the use of the brightness gradient direction

The first described algorithm is based on the Hough transformation. We have developed an algorithm avoiding the use of this transformation because it is known to be time and space consuming and because it is primarily needed to determine only the rotation of the document. To determine the rotation of the document, information on the direction of the brightness gradient of the image is used instead.

The algorithm starts similarly to the previous one. First we reduce the image size but we have to use an averaging algorithm for reduction, which is more time consuming. This algorithm is needed for to preserve the direction of the brightness gradient in the image. Next we enhance the edges with Sobel filtering and calculate the size G and the direction φ_G of the brightness gradient at each pixel. We can define a directional sum of gradient size v – the sum of the gradient size at points with the same direction of the gradient.

$$v(\gamma) = \sum_{i} \sum_{j;\varphi_G(i,j)=\gamma} \left|G(i, j)\right| \qquad (2)$$

The term "same direction" can also mean direction indifferent of rotation of 180°. This directional sum (Figure 9) is very similar to the vertical projection of the Hough transform (Figure 5). Of course, in labeling there is a difference of 90° between the direction of the gradient in the directional sum and the direction of the edges in the Hough transform projection.

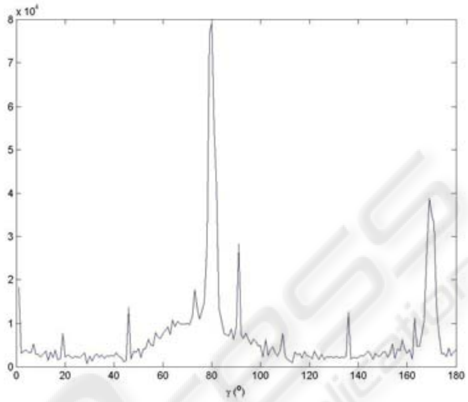

Figure 9: Directional sum of brightness gradient

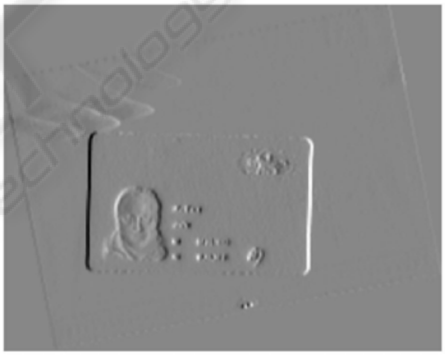

Figure 10: Image of partial derivative with respect to horizontal axis (enhancing vertical edges)

From this sum we can calculate the rotation of the document in the image as we did in the first algorithm. The original scanned image can be rotated in order to get the personal document in horizontal orientation with exactly vertical and horizontal borders. Further processing starts from this new image.

First we reduce the size of the image again and calculate the partial derivatives of the image using the Sobel filter (Figure 10). We can calculate the vertical projection of the partial derivative with respect to the horizontal axis (enhancing the vertical edges) and similarly the horizontal projection of the partial derivative with respect to the vertical axis (Figure 11).

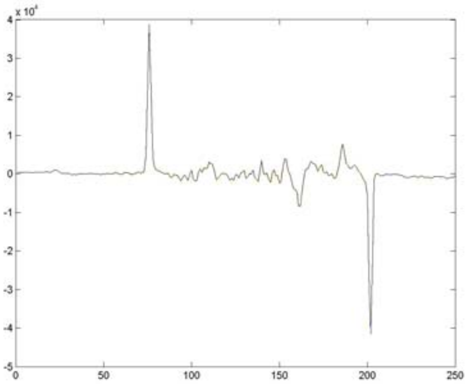

Figure 11: Vertical projection of partial derivative with respect to horizontal axis (compare to Figure 10)

These projections have a maximum at the offset of one border of the document and a minimum at the offset of the parallel border. This gives us enough information to crop the image.

Figure 12: Superimposition of calculated edges on the image of gradient size of a corrupt scan

In the case of the second algorithm we have not discussed the detection of corrupt scans. We calculate this at the end with superimposition of the calculated edges onto the image of the size of the gradient. If the scan is corrupt this superimposition will not cover the edges adequately (Figure 12). Also the ratio between the sizes of the two maxima of the directional sum is returned.

4 TEST RESULTS

The two algorithms were tested from different aspects. First we can look at the results of corrupt scan detection.

Both algorithms return a continuous error estimate, so we have to set a threshold where we declare a scan as corrupt. This limit was set using a training set of images. Next the algorithms were tested using a testing set in which we had 200 good and 36 corrupt scans. For both the learning set and the testing set we determined the corrupt and the good scans by hand. Table 1 shows the test results.

Table 1: Test results for corrupt scan detection

Algorithm with the use of Hough transformation		
	good scan	corrupt scan
alg. detects good	TP: 196 (98%)	FP: 5 (14%)
alg. detects corrupt	FN: 4 (2%)	TN: 31 (86%)
Algorithm with the use of gradient direction		
	good scan	corrupt scan
alg. detects good	TP: 184 (92%)	FP: 7 (19%)
alg. detects corrupt	FN: 16 (8%)	TN: 29 (81%)

From Table 1 we can see that the first algorithm has classification accuracy of 96% (227 correctly classified images) and the second algorithm classification accuracy of 90% (213 correctly classified images).

Another aspect of the algorithms is correct estimation of the document orientation and estimation of document borders. Table 2 shows the test results for estimation of document orientation. When the algorithm was wrong in the estimate of orientation sometimes the error was 90° – algorithm confused the vertical and horizontal edges of the document.

Table 2: Estimation of document orientation

	correct	wrong
alg. Hough	217 (99%)	2 (1%)
alg. gradient direction	218 (98%)	4 (2%)

Table 3 shows the results of estimation of the document border location. When the algorithm was wrong it cropped the document either too tight or too wide and this two types of error are separately shown in the table. Cropping too wide is not as bad as cropping too tight because we do not loose any information – we only have to crop tighter. The test examples that were cropped too wide include many passports in leather-like wrapper, and the algorithm cropped at the border of the wrapper instead of at the border of the document.

Table 3: Estimation of document borders

	correct	too tight	too wide
alg. Hough	183 (81%)	8 (4%)	35 (15%)
alg. gradient direction	189 (84%)	7 (3%)	30 (13%)

The last aspect of the algorithms that was tested was procesing time. The algorithms were tested on an AMD Athlon 1600+ computer with 256 MB

RAM. Algorithm running environment was MATLAB interpreter. The processing time is an average of processing 100 scanned images (Table 4). The processing time does not include the operation of image rotation which is sometimes needed and sometimes not (if the document is already in horizontal position) and it takes about 0.4s to complete. As it is a part of both algorithms it can be omitted from the comparison.

Table 4: Average measured processing time

	average time
alg. Hough	0.26s
alg. gradient direction	0.94s

A very interesting statistic to look at is the profile of the algorithms – proportion of the time that an individual operation consumes compared to the whole algorithm (Table 5).

Table 5: Long lasting operations

Algorithm with the use of Hough transformation	
operation	percent of time
Hough transformation	53 %
Sobel filtering (2x)	23 %
Algorithm with the use of gradient direction	
operation	percent of time
first reduction of size	37 %
directional sum of gradient size	37 %
Sobel filtering (4x)	12 %

The time consumption cannot be directly compared because our implementation of the directional sum of gradient size is much more time consuming than it should be. This is due to interpreter driven MATLAB environment. For a more exact time comparison all the algorithms should be implemented in C for example. The more complicated algorithm for image downsampling in the second algorithm is also standing out.

5 CONCLUSION

In this paper we presented two independent algorithms, which can be used as preprocessing steps in personal document processing applications. They solve the problems of document location and orientation within the image and corrupt scan detection. Some test results are shown that demonstrate the usefulness of the algorithms.

The test results also lead to several ideas of further development in the area of accuracy as well as improvement of the processing time. There is an idea for a two stage Hough transform – in the second stage we could calculate a more detailed Hough transform in a very limited area only (around the edges calculated in the first stage). Multiple corrupt scan identifiers could be used and a machine learning algorithm could determine the good scan – corrupt scan limits. Several operations could be optimized for this problem domain such as image downsampling, image rotation etc.

REFERENCES

Duda, R.O., Hart, P.E., 1972. Use of the Hough Transformation to Detect Lines and Curves. In Communications of the ACM, Vol.15, No. 1

Gonzales, R.C., Woods, R.E., 1992. Digital Image Processing. Addison Wesley.

Horn, B.K.P 1986. Robot Vision. MIT Press.

Hough, V.C., 1962. Methods and Means of Recognising Complex Patterns. US Patent 3069654.

Oppenheim, A.V., Schafer, R.W., 1999. Discrete-time Signal Processing. Prentice Hall, 2nd edition.

2003. ID-Star 4048 documentation. Océ Document Technologies GmbH.