Finger Detection for Quality Assurance of Digitized Image Collections

Roman Graf, Ross King; Austrian Institute of Technology GmbH, Research Area Future Networks and Services, Department Safety & Security; Vienna, Austria

Abstract

This paper presents an approach for automatic detection of fingers that mistakenly appear in scans from digitized image collections. Our goal is to create a reliable detection tool that is independent from scan quality, finger sizes, direction, shape, colour and lighting conditions. Modern image processing techniques are applied for edge detection, local image information extraction, and analysis. We employed expert knowledge to determine default parameters of the algorithm, and support customized parameters for specific institutional workflows. Results for three digital collections analysis are presented. Documents with finger artefacts are identified with high reliability and validated by human visual inspection. The proposed method achieves up to 86 percent classification accuracy.

Introduction

Unintended placement of fingers over the document scan by workers performing the digitization process causes a significant reduction in the quality of the digitized collections. Figure 1 illustrates this problem with sample images taken from various sources.The text obstructed by the fingers is lost and cannot be corrected. Currently no automatic methods exist to detect fingers on scans and a human expert involvement is required to detect corrupted files in digital collections. However, the increasing number of digitization projects world wide, the number of pages involved, and the associated economic investments require the development of automated quality assurance solutions.

This paper reports on the development of an automatic method for the analysis of digital document collections, for reasoning about analyzed data and for decision support regarding finger detection. We aim at designing a robust method for automatic finger detection on scans and for subsequent decision making support in order to increase the quality of digital collections.

The paper is structured as follows: Section 2 gives an overview of related work and concepts. Section 3 explains the finger detection process and also covers parameter identification and image processing issues. Section 4 presents the experimental setup, applied methods and results. Section 5 concludes the paper and gives outlook on planned future work.

Related Work

The challenges of finger detection are caused by varying image quality, different finger sizes, direction, shape, colour and light conditions. Currently employed related techniques are trained-model, vision-, or colour-based. None of them completely address the given challenge.

Finger masking using closest similar zones of the background images based on a model was described in [1]. However, this method appears to be applicable only for finger detected in the near from border and the image processing method is not further specified. In many real life cases fingers cannot be modeled due to large variations in shape, size and illumination. Sometimes a finger only appears in an image as a light spot or as a blurred shape. This is one reason why the Support Vector Machine method described in [2] cannot be effectively applied to finger detection. This approach makes use of multi-scale deformable part models for person and vehicle detection. The model represents the visual appearance of highly variable objects but requires a rich training set with partially labeled data. In contrast, our approach concentrates on common finger shapes and does not require elaborate training data.

In a multi-resolution approach for page segmentation described in [3], the page is broken down into several blocks in order to separate content in to text, drawings, and pictures. In order to achieve this, the authors apply varying levels of pixel intensity: average, median, variance and threshold. The drawback of this method is that finger detection requires additional analysis of the extracted picture areas in order to separate fingers from other objects.

In [4], the authors present a framework for interaction by hand gestures in which the hand detection is based on a cascaded detector trained on a specific initialization gesture. The segmentation is performed using an adaptive 3D modeling of the hand colour. When applied to finger detection on scans, colour-based methods lack accuracy due to varying skin colours or the use of rubber gloves in multiple colours, or the use of gray scale images and image quality.

Finger Detection Process

This work introduces a new approach for the reliable detection of scans that are corrupted by unwanted finger depiction in a digital document collection within the scope of a quality control

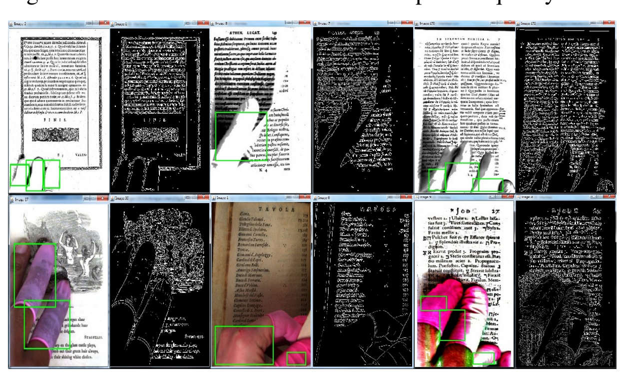

Figure 1. *The sample positive detections.*

process. The analysis is based on image processing methods and the OpenIMAJ library [5] for edge detection. The proposed approach allows the end user to customize the analysis parameters but does not require expertise in the image processing domain.

However, the basis for accurate finger detection is information aggregated from digital documents and from experimental knowledge provided by human experts. A process of decision making for image quality assurance in digital preservation requires expert knowledge of image processing and library processes.

In order to facilitate the management of the finger detection method, the expert knowledge was integrated in the quality assurance workflow shown in Figure 2. The user triggers a complete collection analysis. The results of analysis are presented in finger candidate images (see Figure 1) where suspected areas are marked by green rectangles. In this way the user is able to draw a conclusion as to whether a finger candidate image is actually a finger.

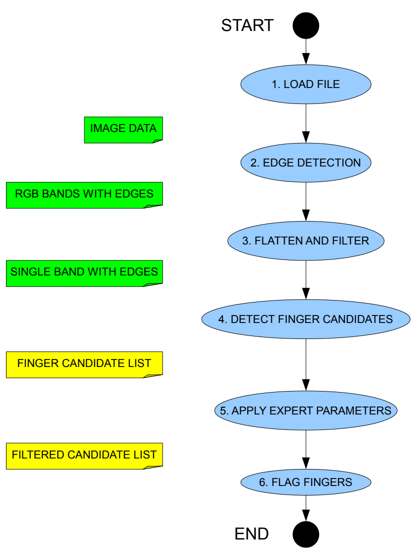

Figure 2. *The finger detection workflow.*

Suggested Method and Parameter Identification

Information retrieved from the digital document is processed by the customized domain model. This model enables structured handling of analyzed data and applying of parameters, which have been obtained from the domain expert of digital preservation and from conducted experiments.

Figure 3 depicts the main expert parameters involved in the finger detection algorithm. The points a and b represent points on the finger edges that have the same coordinate value on the Y axis. The distance between this points is a minimal finger width W_{min} for the given Y coordinate. The X_b parameter represents the gap between the finger edge and the page border. The A_{vg} parameter stands for the average distance between finger edges for all Y coordinates associated with this finger candidate. The dashed line S represents a threshold where the finger shape is cut in order to facilitate calculation and to reduce the number of false positive results. The L parameter fixes the maximal accepted finger size.

The default parameter set (Figure 3) comprises:

Pixel value threshold P. The pixel value is a value in range 0.0 to 1.0 in the flattened single band grayscale image. The pixel value threshold defines a threshold at which this pixel should be taken into account for further analysis.
Minimal finger width $W_{n\ m}$. This parameter describes the minimal pixel count that should contain a finger candidate on the X axis.
Average width rate $A_{n\ m}$. This parameter is used to set up an acceptable threshold for the thinnest and the thickest distance between finger boundaries in order to follow a relatively smooth finger shape without large volatile changes.
Pixel variance $_{n\ m}$. Due to image distortions often it is not possible to evaluate an accurate line. Therefore we have to set up acceptable variance for neighboring pixels that could build a line with a current pixel at analysis.
Minimal finger points F_{min}. In this parameter we define the minimum number of pixel points regarded as detected pixels for a finger candidate.
Maximal finger size $L_{n\ m}$. This parameter defines the maximum size of finger candidate.
Minimal border distances x_b y_b. In order to involve page border factor in our calculations we define a minimal page border distances for both axes.

The default parameters provide good accuracy, but users can adjust parameters like minimum and maximum finger size, variance, minimum number of finger points and the distance from border in pixels in order to tune their workflows.

The finger detection algorithm is summarized as follows:

1. The workflow starts with loading of data from given file.
2. Since the point- and edge-based features are regarded to be robust against lighting variants [6], the Canny edge detection algorithm is employed [7] in order to retrieve edges of a RGB image. Additionally at this stage we flatten the bands of the original image, using the average value of the pixels at each location, in order to obtain a new single-band image.
3. We filter the flattened single-band image pixels by applying a threshold parameter P for pixel value, where P is a coefficient with value in the range from 0.0 to 1.0.

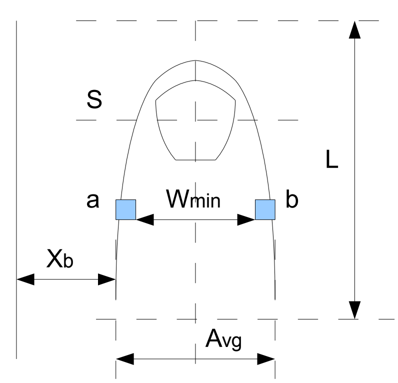

Figure 3. *The parameter used in finger detection algorithm.*

4. We analyze a given scan image in order to detect finger candidates with associated pixel coordinates. To do so we evaluate the next existing pixel on the right from the given position on the X axis and examine its coordinates regarding association with current finger candidate.
5. Given a preliminary finger candidates list, we apply the parameters evaluated from expert knowledge like finger size, points count, variance and distance to border to evaluate the extent to which the candidate matches the requirements.
6. In the final step we draw green rectangle around the detected finger areas and display the resulting image for user reasoning.

Image Processing and Algorithmic Details

As noted above, we apply the Canny edge detection algorithm in our image processing workflow in order to extract the significant shapes from the image and to extract a gray scale image with additional filtering. We analyze extracted shapes and isolate finger candidates. In our calculation algorithm we use logic similar to that applied in the stroke width transform method implemented by [8], which is used for text recognition. For each pixel we investigate pixels in the neighbouring area and follow lines according to given expert parameters. In this way we create a finger object and if it matches all conditions defined for a finger we mark the evaluated region as a detected finger by green rectangle to facilitate further analysis for an expert.

The coordinates for the finger candidate structure are computed using an algorithm for finger pixel calculation (see Equation 1).

$$x \in 1\ \ M \qquad (1)$$

$$y \in 1\ \ N$$

$$F(x\ \ y) = \begin{cases} 1 & \text{if } f(x\ \ y) > P \\ 0 & \text{else} f(x\ \ y) \leq P \end{cases}$$

$$G(x\ \ y) = \begin{cases} F(x\ \ y) & \text{if } (x > x_b) \wedge (x < M - x_b) \wedge (y > y_b) \\ \wedge (y < N - y_b) & \\ 0 & \text{else} \end{cases}$$

Here G represents the pixels in gray scale image after the edge detection step, computed over the image dimension, $F(x\ \ y)$ represents the finger coordinate points set depending on $f(x\ \ y)$ function and pixel threshold P value. N and M represent the initial pixel coordinates of a finger candidate for the X and Y axes. Several constants were required for accurate computation: x_b and y_b stand for the offsets from X and Y axis with respect to the image edge.

A pixel count around an initial point of a finger candidate is computed as a sum over all evaluated and filtered pixels, which are located in acceptable distance to the initial point.

Evaluation

This section describes experiments that were conducted using our finger detection algorithm. The code was written in Java and executed on an Intel Core 2 Duo T9600 (2.80GHz) computer.

Collection Analysis

In the performed experiment we analyzed three digital collections. The goal of the evaluation is a reliable and accurate detection of scans with unwanted finger images. Our hypothesis is that the reported approach should be able to detect corrupted documents with good reliability and to ignore unattended scans. Scans that are flagged by finger detection algorithm should be additionally analyzed by human expert.

All analysed scans were retrieved from the Internet. We manually created ground truth data for the assessment of the evaluation results. One collection contains 160 corrupted images that include fingers. The second collection comprises 26 images with fingers. The third collection contains 730 images and is a reference collection without corrupted documents. This collection is used in order to ensure that a simple text document scan without fingers on it will not be detected by our algorithm as a false positive.

The result of the analysis is a set of documents with finger candidates marked with a green rectangle (see Figure 1). The proposed method has been tested with a variety of images from different origins using a default set of parameters. Mis-classifications coming along with correct results are few and happen with shapes similar to finger shape definition. Improvements in the algorithm and filtering of mis-classifications are subjects of a future work.

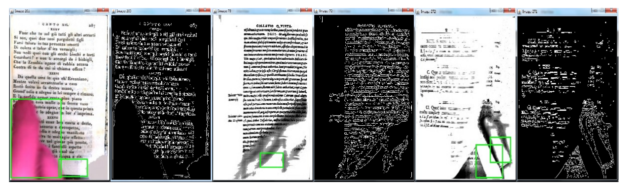

Figure 4. *The sample for correct detection of blurred images.*

As shown in Figure 4 even blurred fingers could be correctly detected by the algorithm.

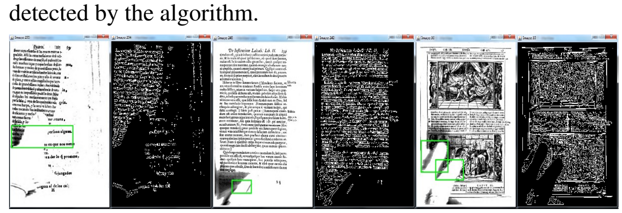

Figure 5. *The sample for correct detection of fingers that are looking as a light spot on the scan surface.*

Some fingers like those in Figure 5 were correctly detected despite the fact that they appear merely as a light spot on the scan surface. Due to the typical finger shape, it was still possible for the algorithm to correctly detect them.

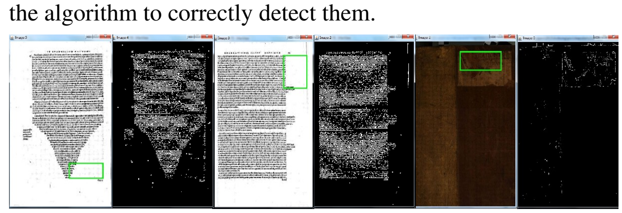

Figure 6. *The sample false positive detections.*

Figure 6 illustrates false positive scans detected in the third collection. The reason for these false detections is that shapes of the text, warping shadow, and cover surface inconsistencies create a structure that looks similar to the finger form. Such cases are difficult to avoid and manual expertise is required to eliminate these false positives.

The digital documents presented in Figure 7 were not detected by our algorithm. The reasons are that these fingers are strongly blurred, are obscured by shadows, or only tips of the fingers are depicted.

Effectiveness of the Detection Algorithm

In the first collection, fingers were correctly detected on 139 scans. In the second collection, we evaluated 20 correctly detected scans from a total 26 scans that should be detected. In the third collection, only three images were falsely identified as containing finger depictions. Therefore the effectiveness of our detection algorithm, according to the Relative Operating Characteristic (ROC) method, is measured by the distribution of collection points (0,0.8633), (0,0.7692) and (0,0.9958) in ROC space. These are quite good classification results that are very close to the best possible classification point (0,1).

The total time for finger detection for the scans from the considered collection takes 185767, 8915, and 262363 seconds for the first, second and third collection respectively. We correctly evaluated corrupted images in up to 86 percent of cases in the test collections. The experimental results demonstrate that our approach is very promising for making the digitization process more reliable and for ensuring the quality of digital collections.

Conclusion

We have presented an approach for automatic detection of finger in digitized image collections, for reasoning about analyzed data and for decision support regarding finger detection. We applied modern image processing techniques for edge detection, local image information extraction and its analysis for reasoning on scan quality.

An important contribution of this paper is the creation of automatic and reliable finger detection tool that is independent from scan quality, finger sizes, direction, shape, colour and lighting conditions. We employed expert knowledge of scanning and library processes to define parameters for our detection algorithm.

The experimental evaluation presented in this paper demonstrates the effectiveness of employing the suggested image processing techniques for generating reasoned suggestions. With up to 86 percent classification accuracy, the tool reliably detects documents containing finger what was demonstrated by human visual inspection. An automatic approach delivers a significant improvement in terms of personnel costs when compared to manual analysis.

The analysis tool for digital collections presented in this paper can help to ensure the quality of digitized collections and supports managers of libraries and archives with regard to long-term digital preservation. As future work, we plan to improve the automatic quality assurance approach of the image analysis by improving the accuracy of finger detection and by reducing the parameter count.

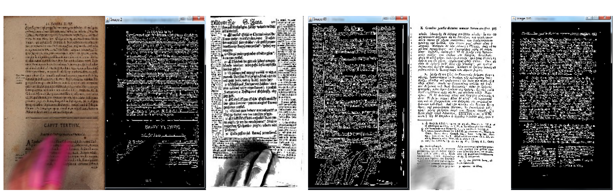

Figure 7. *The sample for scans with not detected fingers.*

Acknowledgments

This work was partially supported by the SCAPE Project. The SCAPE project is co-funded by the European Union under FP7 ICT-2009.4.1 (Grant Agreement number 270137).

References

[1] Le Bourgeois, F., Trinh, E., Allier, B., Eglin, V., Emptoz, H., Document images analysis solutions for digital libraries, Document Image Analysis for Libraries, Proceedings of the First International Workshop on Document Image Analysis for Libraries (DIAL 2004), IEEE Computer Society Washington, DC, USA, pg. 2 - 24, 2004.

[2] Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D., Object Detection with Discriminatively Trained Part-Based Models, Pattern Analysis and Machine Intelligence, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.32, no.9, Chicago, IL, USA, pg. 1627-1645, Sept. 2010.

[3] L. Cinque, L. Lombardi, G. Manzini, A multiresolution approach for page segmentation, Pattern Recognition Letters, Volume 19, Issue 2, February 1998, pg. 217-225.

[4] Wagner, S., Alefs, B., Picus, C., Framework for a portable gesture interface, Automatic Face and Gesture Recognition, 2006. FGR 2006. 7th International Conference on Automatic Face and Gesture Recognition, pg. 275-280, 2-6 April 2006.

[5] Jonathon S. Hare, Sina Samangooei, David P. Dupplaw, OpenIMAJ and ImageTerrier: Java Libraries and Tools for Scalable Multimedia Analysis and Indexing of Images, Proceedings of the 19th ACM international conference on Multimedia, November 28 - December 1, 2011, Scottsdale, Arizona, USA, pg. 691-694.

[6] Marr, D., Hildreth, E., Theory of edge detection, *Proc. of the Royal Soc. London*, pg. 187 - 217 (1980).

[7] Canny, J., A computational approach to edge detection, *IEEE Trans. Pat. Anal. Mach. Intell.*, pg. 679 - 698 (1986).

[8] Epshtein, B., Ofek, E., Wexler, Y., Detecting text in natural scenes with stroke width transform, Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on , vol., no., pg. 2963-2970, 13-18 June 2010.

Author Biography

Roman Graf is currently a researcher for the EU SCAPE project at the AIT in the Digital Memory Engineering group. His current research interests are in digital preservation and image processing. He received his Ph.D. in electrical engineering from Riga Technical University.

Ross King received his Ph.D. in physics from Stanford University. After moving to Vienna in 1995, he migrated to the IT sector, and joined AIT in 2002 to help found the Digital Memory Engineering working group, which he presently leads as Senior Scientist and Thematic Coordinator for Next Generation Content Management Systems. Dr. King serves as board member and financial officer of the Open Planets Foundation and he is also the Coordinator of the FP7 ICT project SCAPE. His research interests are primarily concerned with multimedia information management and digital preservation.