PREPROCESSING PHOTOS OF RECEIPTS FOR RECOGNITION

Wojciech Korobacz, Marek Tabędzki

Faculty of Computer Science, Bialystok University of Technology, Białystok, Poland

Abstract: The subject of this work is methods of image pre-processing, applied to receipts photos. The purpose is to improve their quality, allowing to increase the efficiency of the conventional text recognition software (OCR). The authors had mainly difficult cases in mind – photos taken freehand in unfavorable lighting conditions. The work describes the analyzed methods of filtering, binarization, searching for the edge of the image, image straightening, marking the area of interest, thinning. The preliminary results with OCR software on a small data set were also presented. Thanks to pre-processing, character recognition efficiency has been improved by 25%. The final part presents conclusions and plans for future work.

Keywords: digital image processing, optical character recognition

1. Introduction

The subject of this work is a method of pre-processing of photographs containing receipts to a form readable by conventional text recognition software (OCR). This is an important issue from the point of view of technology development, in particular mobile and interactive technologies. The smartphone has become the object of everyday use these days. A portable computer with a camera and Internet access gives almost unlimited potential for practically every area of our lives. Only a few years ago, the peak of the possibilities was a telephone that, apart from calling and sending messages, was able to take pictures or store notes. Modern smartphones are used as handheld instant messengers, cameras, alarm clocks, schedules, information and news sources, gaming centers and much more. Today, we want to have everything on the smartphone that will make life easier for us. This idea is in line with the latest IoT trends "Internet of Things" or even newer IoE "Internet of Everything".

Following this idea, the smartphone has also become an e-wallet, with the help of NFC (Near Field Communication) we can, for example, pay for purchases and

Advances in Computer Science Research, vol. 14, pp. 87-103, 2018. DOI: 10.24427/acsr-2018-vol14-0006

collect data about expenses. However, only the amount can be drawn from the payment information, but not the content of our purchases. The fastest and easiest way to achieve this goal is to take a photo of the receipt using the application on our smartphone, which will properly process the photo and create a text file with information from the receipt. Then, this information should be processed, that is, to extract relevant information from unnecessary information, group them and save in a way that allows later analysis.

The problems that the following work deals with are the randomness and uniqueness of photographs, i.e. inhomogeneous lighting conditions, cropping, different angles of images taken, non-linear distortions and sharpness of images. For such conditions, it is necessary to analyze, study and select the best algorithms of image processing in such a way that the final result is as close as possible to the later readout by OCR software.

This issue is not new. With the appearance and popularization of smartphones, researchers considered the possibility of using their cameras to scan documents and read text. The quality of this type of material has always been a problem, significantly differing from the images obtained with the use of a flatbed scanner (usually expected by the OCR software). The influence of the environment resulted in degradation of quality, lowering readability. Therefore, the work devoted to preprocessing techniques and quality improvement, eg Sharma's work [13] treats about the impact of noise and blur, while Bieniecki in his work [1] emphasizes the correction of perspective and orientation. As a confirmation of the effectiveness of the proposed methods, likewise the authors of this paper, he uses existing OCR software. A similar direction was chosen by Shen [14], however, focusing on removing the background that could disrupt, thus negatively affect the process of text recognition – the answer is the appropriate thresholding procedure, which is able to get rid of the unwanted background image. Research shows that a properly selected pre-processing procedure is able to improve the quality of recognition, eg Wiraatmaja in his work [18] noted a quality increase of more than 25% compared to the standard approach, while Brisinello [2] in his low-resolution and low-quality image studies, for which simple OCR software provided only 35% of recognition efficiency, improved quality through preprocessing procedures by over 33%.

This publication presents the results of research carried out as part of the master's thesis of one of the authors [5]. Presented work explores available solutions and algorithms in the field of digital image processing. It does not deal with the next stage, which is the identification of data read from the image. Therefore, it should be treated as a preliminary to further research, which will allow to create a full system implementing all of the tasks described above.

2. Problem and solution concept

To select the appropriate samples for testing, the classes of photo receipts we can deal with were determined. The following characteristics of the samples were considered:

- Cropping – whether the entire receipt is visible, how much background is in the picture,
- Lighting – it can be artificial or natural, strong or weak, shadows can be seen on the receipt,
- Sharpness – whether the photo is sharp or blurred,
- Angle of rotation – how much the photo deviates from the vertical position,
- Folds – the receipt may be curled or folded.

On this basis, a database of samples for research was collected (some of them are shown in Figure 1), remembering their diversity and taking care of the representation of each of the mentioned classes.

Fig. 1. Examples of collected data

In total, about 240 samples were collected – photographs of various receipts taken freehand using smartphone. All receipts were in Polish, but they could be different in font, content layout, shade and dimensions. It was ensured that the picture contained a receipt of at least 20% of the surface. Because the pictures were taken using different cameras, they could have different resolutions. Collected samples were grouped in 9 visually similar classes (in terms of the above-mentioned features), and then the subsequent stages were evaluated by researchers separately in each group.

The following test run was specified:

- Pre-processing: image conversion to grayscale, filtering, contrast enhancement,
- Binarization: conversion to black and white,
- Finding the receipt edge in the image,
- Straighten: rotate to a form in which the text is in a horizontal position,
- Finding the area of interest: location of the text in the picture,
- Thinning: changing the text to one pixel wide form,
- Tests of processed images in OCR programs.

The above steps to prepare a picture and read the text from it were considered in turn, and then the final solution was formulated, which will be presented further. The pre-processing stages were subjected to preliminary visual assessment. Often it was easy to assess whether the choice of methods or parameters improves the quality of the image or causes its further degradation. On this basis the most promising method was indicated and passed to further stages, including final experiments with OCR software.

The Java programming language was used for the implementation and the algorithms available in the open-source OpenCV library were used.

3. Filtering

The first step in processing the photo of the receipt was its conversion to a gray scale image. This operation does not require a specific description, as the OpenCV library procedures have been used. The image was converted from a 24-bit to an 8-bit form according to the following equation.

$$Y = 0.299 \cdot R + 0.587 \cdot G + 0.114 \cdot B \tag{1}$$

Where R, G and B are red, green and blue intensities, and Y is the output intensity.

A more important element of the pre-processing was smoothing and histogram equalization. Smoothing operation reduces noise in the image (but also blurs the focus). It was implemented using a Gauss 5×5 filter. The histogram shows the distribution of the brightness values in the image [6,4]. The operation of histogram equalization (flattening) improves the contrast by extending the histogram to the full gray scale and spreading out the most frequent intensity values.

In the next steps, both filtered and unfiltered images were taken into consideration to assess the impact of the pre-processing on the result.

3.1 Binarization

Image binarization (or thresholding), is changing the image from the gray scale, usually 8-bit, into an image of two intensities, white and black [9]. This operation accomplishes the task of segmentation – it separates the subject of interest (text, in the case of receipts) from the background in the examined image.

Binarization methods can be divided into three categories: global, local and adaptive [6]. In global methods, one threshold value is set for the whole image, based on which it decides whether a pixel of a given brightness belongs to the text or the background. This is a relatively simple task in the case of clear, good-quality photos with high contrast between the text and the background. In more problematic cases, it is necessary to use the local methods or adaptive ones. In local methods, the threshold value is determined for a fixed-size window, while in adaptive ones, it is determined separately for each pixel.

Three binarization methods were examined (Eq. 2). The first one was the classic global Otsu method [7]. It is based on a discriminant analysis. The threshold is determined on the basis of minimizing intra-class variance and maximizing inter-class variance. The other two methods were adaptive methods [4]. In methods of this type, an important parameter is the distribution of the influence of neighbors in the observation window on the examined pixel. This is shown in the following equation:

$$Y'_{x,y} = \begin{cases} 1 & \text{if } Y_{x,y} > T_{x,y} - C \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

where $Y_{x,y}$ is a source pixel intensity, $Y'_{x,y}$ is destination intensity (zero-one value), $T_{x,y}$ is a threshold calculated individually for each pixel, and C is a equivalent of the coefficient k in simplified Niblack method [17]. For the first method, the threshold value T is a mean of the pixel intensities in the observation window. For the second one, it is a weighted sum (cross-correlation with a Gaussian window) of this neighborhood.

The second parameter examined was the size of the observation window. The size of the window was dependent on the size of the input image and two versions were tested for each method: 1% and 3% of the width of the image being examined (larger windows were rejected due to blurring and merging of letters). In addition, as described earlier, versions with and without histogram equalization and smoothing were examined. In the case of smoothing with a Gauss filter, different values of the σ sigma factor (determining the weight of subsequent neighbors in the mask) were considered. Selected results are presented in Figure 2.

Visual evaluation of the processed images led the authors to the conclusion that Otsu's global method copes well with clear, sharp images with a good lighting. Unfortunately, with the inferior input material, this method fails. In the adaptive methods

a b c d

Fig. 2. Sample image (a) and same image after binarization: with Otsu method (b), adaptive method with 3% window (c), adaptive method with $C = 5$ (d)

it can be observed that along with the growing observation window, the text becomes less readable, which should be considered an undesirable situation, as it will negatively affect the ability to recognize the text.

Both alignment and smoothing of the histogram have a significant impact on improving the readability of the image for humans, but it does not equal the positive impact on the binarization or any further recognition process. In particular, the histogram equalization, due to the loss of some information, introduced disturbances and caused problems in the binarization. In the case of smoothing, only for low sigma values this has a positive effect on the result.

Because the observations showed that it is not possible to indicate the best solution for each type of photo, an additional factor C was introduced in the adaptive methods (equation 2). Subsequent tests, carried out on the photographs with and without smoothing (Figure 2d), allowed to obtain the highest quality for the $C = 5$ and an observation window of a 1% width of the input image (in the adaptive method with a homogeneous neighbors weight distribution) and a 3% window (for the adaptive method with the Gaussian distribution). A larger window of observation reduces the impact of noise, but also blurs the text. In turn, too small a window, adversely affects the continuity of letters.

Comparing visually, the adaptive method with the equal weights in the observation window is the best, but full tests along with the subsequent stages will give the final verdict.

Several tests were carried out to evaluate the processing time and they showed that the differences between the individual methods are significant. The Otsu method usually dealt with image processing within 12-14 ms, the adaptive method with equal weights usually took 32-37 ms, while the most time-consuming method is the adaptive method with neighbors weights Gauss distribution, which needed over 100 ms for an observation window of 1% of the image width and more than 300 ms for a 3% wide window, so the time consuming increased with the increase of the number of neighboring pixels to be tested.

It should be noted that these tests were performed on a desktop computer (Intel Core i5-520M processor, 4GB RAM). Tests on a real mobile device will be carried out in the future. It will then be possible to assess whether the processing time should be the decisive factor when choosing the method. The mentioned processing time is always an additional overhead, perhaps, however, it will become an irrelevant factor in the duration of the entire process. It may also turn out that a properly selected preprocessing approach will shorten the time of image analysis by the OCR module. This will become the subject of further research.

3.2 Edge detection

The next tested process was edge detection. Its goal was to find the edge of the receipt, to be able to determine its rotation and determine the necessity of straightening (and also find the right angle).

It was decided to implement this with the help of Canny's well-known and proven method [3]. Attempts to process photos without prior processing showed that the resulting image is too noisy for further work, therefore pre-smoothing the image with a low-pass filter is necessary. Two different Gauss smoothing filters were adopted in the study, with coefficients such as in the examination of the binarization methods (observation window 1% of the image width and sigma $\sigma = 2$ and observation window 2% of the image width and sigma $\sigma = 3$). In addition, different threshold values in hysteresis were tested.

The best results were obtained with less smoothing (smaller window and sigma coefficient), for thresholds $TD = 100$ and $TH = 120$ (Figure 3b). At lower threshold values, the results are noisy, while at higher ones, information is lost.

For comparison, the method of the morphological gradient was also tested (Figure 3c). This is the difference in morphological dilatation and erosion [9]. Various shapes of the 3×3 structural element (square, round and cross) were used.

Unfortunately, taking into account the basic goal – determining the receipt angle – both of the tested methods proved to be insufficient. Therefore, it was decided to propose another solution for finding the contours of the receipt.

a b c

Fig. 3. Original image (a), Canny edge detector (b), morphological filter (c)

3.3 Rotation angle detection

Due to the unsatisfactory results of the edge detection algorithms, it was decided to test other approaches, this time with only the need to find the right angle, allowing for correction of receipt rotation.

The first of the approaches considered is based on the use of Hough transform. Hough transform allows you to find specific shapes in the image. Mostly these are lines, circles, but there are also solutions that allow you to search for more complex shapes. Its advantage is that it works even when the shapes do not maintain continuity. A detailed description of the method can be found in [8,4]. Its idea is to transfer individual pixels into the Hough space described with the shape parameters that we want to detect. In this way, for example, the line in the input image will be represented by a point in Hough space. When looking for the maximum values in Hough's space, we can find the location of the lines in the input image.

The prepared image had to be binarized in such a way that it contained as little noise as possible. From the point of view of this method, the sharpness and legibility of the detected elements were an irrelevant parameter. Thus, mathematical morphology (gradient method) was used. Then, in the iterative process, using the Hough transformation, lines that met the length criterion were searched for in the image. In the first step, a line that was going through the entire image was searched. If it was not detected, a line twice as short was searched, which was then repeated in the loop until one or more lines were found. For the found lines, their angle of inclination was calculated in relation to the vertical – their average was the angle at which the image should be rotated (Figure 4).

a b

Fig. 4. Input image (a), rotated image (b)

Experiments with images have shown that the above algorithm is perfectly suited for detecting the angle of rotation in situations where a photo of the entire receipt is in the frame, i.e. the outer edges of the receipt are visible. In a situation where there is only the content of the receipt in the frame, the algorithm is able to detect only the lines of the largest letters. The problem was also when the receipt was wrapped or folded.

Another tested algorithm was the original idea of looking for an angle based on the course of the text lines in the image. The first step was the binarization and denoising of the input image. The results obtained in the previous steps were used and the Gaussian filter with the size of 2% of the image width and $\sigma = 3$ was selected and the adaptive thresholding method with equal neighbors weighed, the observation window being 1% of the image width and the $C = 5$ coefficient. Then, in the loop for angles from -10 to 10 degrees, a vertical histogram of the image is calculated. For each angle tested, the sum of the pixel values of each line is calculated. On the basis of the average brightness, the lines referred to as “white” (with the highest brightness with a threshold of 30%) are selected. The angle for which the number of "white" lines was the highest is the sought angle.

Although the proposed method in many cases proved to be effective, we also found the classes of images for which it failed. The problem was the cases when a large part of the image was taken up by the background. It works best on images containing the text itself, where the angle was detected correctly. This created the need to develop a more effective method.

3.4 Text outline detection

In the next stage of the research, a method of universal, automated detection of the text outline was proposed based on existing solutions. It consists of three steps:

1. Pre-rotating the image
2. Detecting the entire text outline and marking it in the image
3. Cutting out the image of the smallest area containing the entire text

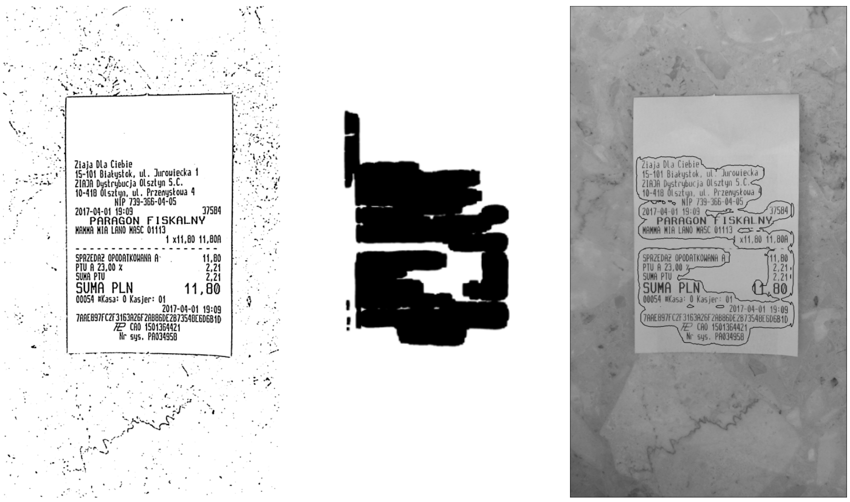
Image rotation The first step was to pre-rotate the picture to the correct position. For this purpose, the Hough transform was used as described in the previous chapter. Rotation is important because otherwise, even if the area with text is found correctly, it can be cut with the background.

Outline detection The second stage was to find the position of the text with the rejection of an unnecessary background. For this purpose, two algorithms were used.

The first one was based on Canny's edge detection algorithm. Its course was consistent with the description in the previous part of the work – i.e. with binarization, the window of observation 5% of the width of the image and the coefficient $C = 5$. Then, the morphological operation of erosion with the element 3×3 was used. This led to merging the characters of the text into one blot (Figure 5).

The second of the considered algorithms was based on a high-pass filter based on edge detection using the Sobel operator. This is one of the convolution filters specializing in edge detection [12]. Due to its properties it is quite resistant to noise. The remaining steps were the same as above – until the text was merged. The only difference was that the erosion was carried out in the loop ten times, because once erosion resulted in insufficient merging.

Image cropping The third stage was the same, regardless of the previously chosen methods. It assumed an input of a with the text in the form of merged blots. At this



a b c

Fig. 5. Sample binarized image (a), after erosion (b), shape marked on original image (c)

stage, it was necessary to correctly find these blots and then cut out the smallest fragment containing them from the image.

The first step was to find all the contours based on the input image. Due to the optimization of operation, the simplified representation of contours was taken.

Then, the found outlines were filtered out. Because the algorithm finds the contours of even the smallest noise, the found outlines with an area of less than 0.1% of the area of the input image were discarded. Contours larger than 95% of the image area were also discarded (to eliminate the outline of the whole image). All contours whose surface area contained within this interval were taken into account as significant from the point of view of the described algorithm.

Then, the rectangles escribed on the given contours were found. Next, the rectangle escribed on the whole set of contours was searched for (Figure 6).

It was accomplished by simple algorithm that finds the minimum rectangle containing a set of rectangles. It was only necessary to make sure that the found rectangle did not go beyond the size of the original image. On this basis, a smaller image with dimensions matching the found contour of the text was cut from the original image.

The experiments on the collected samples allowed us to observe that better results are obtained by an approach based on the Canny algorithm, because it found a contour closer to the content of the receipt shown in the pictures.

a b c

Fig. 6. Image with found shapes marked (a), bounding rectangles (b), cropped image (c)

3.5 Thinning

The task of the thinning stage is to represent the shapes shown in the image with the form of lines with a width of one pixel (Figure 7). This is helpful and sometimes even required in some text recognition methods. Some OCR software can do it on its own, while some can work better without it (if it is based on the full shape of the letter, not the lines themselves).

a b c

Fig. 7. Input image (a), image after binarization (b), image after skeletonization (c)

The work did not test different methods of thinning, however, based on previous experience of the authors, they propose the K3M skeletonization algorithm [10,16]. It is a fast and effective method that gives visually good results and has proven itself in many related applications. The influence of skeletonization on the OCR read will be discussed in the next chapter.

4. Optical Character Recognition

The final purpose of the procedure of processing a receipt photo is to recognize the text located in it. Therefore, it was necessary to test obtained images trying to identify the text located on them and in this way to assess the effectiveness of the proposed techniques. As part of this work, the original recognition system was not implemented, but available implementations were used. The first of them is the open-source Tess4j[1] library and the ITesseract class it offers. It is a simple and not flawless method, but it was decided to use it for comparative purposes.

The second application was the commercial ABBYY Fine Reader[2] application, one of the most popular applications of this type. The work uses the available trial version, which allowed for 50 readings, which was insufficient for the work presented. For this reason, it was decided to limit the size of the data set.

Comparison of these implementations on sample images allowed to conclude that the Fine Reader application gives much better results. Due to the fact that the program is intended for processing scanned documents, it worked best with images that were free from noise, shadows, distortions, and the text was clear and legible in them. It is worth noting that it did not require a thinned text – and even gave worse results after thinning. This clearly indicates that when using this program this stage should be omitted.

Another conclusion from the analysis of results is that more difficult cases of receipt images may require a dedicated solution, specially prepared for this type of image, and the image processing itself may be insufficient.

4.1 Comparing OCR results before and after pre-processing

The following results were obtained using the ABBYY Fine Reader OCR application. One representative of each of 9 the classes mentioned earlier (section 2) was selected

[1] `http://tess4j.sourceforge.net/`
[2] `https://www.abbyy.com/en-ee/finereader/`

randomly for study. The effectiveness of text recognition in the pictures before and after the pre-processing was compared.

Table 1 presents the results obtained. The correctly identified characters and words were calculated for individual samples. The results may seem low, as for the OCR system, but it is worth paying attention to several reasons. First of all, the selection of the samples – many of them were difficult and problematic. In addition, the method of assessing the correct diagnosis was quite rigorous and refused a character or word in the event of the slightest deviation. In the final solution, one should use the semantic analysis to increase the level of identification by introducing a dictionary or knowledge about what characters or words we expect in a given part of the receipt. Here it was not implemented. Many errors most probably resulted from the use of specific typefaces in part of cash registers, which apparently were not in the ABBYY Fine Reader data base, because some characters could not be identified in any instance. Here, it would certainly help to teach a system of specific fonts, used by the most popular cash registers in a given region. At the same time, these results are similar to the results reported by other authors (as described in section 1).

Table 1. A comparison of the effectiveness of character recognition in images before and after processing

	average	minimum	maximum
Words recognized in original images	**24%**	1%	56%
Words recognized in pre-processed images	**33%**	1%	62%
Characters recognized in original images	**41%**	1%	76%
Characters recognized in pre-processed images	**51%**	3%	82%

Careful analysis of the results for individual samples also revealed that for some of the examined photos the level of recognition was extremely low – around 3%. These were photos that were heavily corrupted (e.g. by uneven shade) and even the use of a processing algorithm had little effect on the result. These cases have been identified and will be analyzed more closely in subsequent work.

The results presented were made on a small data set, which is why they can not be considered as proving the effectiveness of the discussed techniques. For a proper assessment, it should be repeated on a full data set. However, it can be concluded that the approach is promising and it is worth continuing research. Text on pre-processed images has been recognized with a higher efficiency than on those before processing. The text recognition efficiency has been improved by an average of 25% in the case of characters and over 35% considering the entire words. Importantly – in each of the cases studied, the result improved by using the proposed processing algorithm. These results are comparable to the results of other authors (as mentioned in section 1).

5. Conclusions

After analyzing various pre-processing methods and examining the results obtained by OCR applications on the collected examples database, the following algorithms have been proposed:

- Straighten the image by locating the angle of rotation using the Hough transformation
- Finding a text outline using the described algorithm using the Canny method
- Image binarization by adaptive method

Its application allowed to obtain optimal and reproducible results on the tested sample. The vast majority of samples, regardless of the lighting conditions, cropping, angle of rotation or size, has been processed in a satisfactory manner. The output images are similar, comparable and reproducible. Each is aligned, the text takes up most of the frame and noise has been reduced.

Research using the OCR system was performed on a small database of samples and should be treated as preliminary. Nevertheless, they are promising and encouraging further research – character recognition efficiency has been improved by 25%. In the future, the authors plan to apply algorithms on a larger data set, in order to validate the approach. The direction of further work will include also the identified problematic receipt classes and try to address these problems. The solution may be to improve certain stages of the process, eg it is planned to use the binarization method proposed by Sauvola [11], intended to deal with noise, uneven lighting and other types of degradation of image quality. In addition, it is planned to use own implementation of the module recognizing the text, based on the authors' experience in the field of word recognition [15].

References

[1] Wojciech Bieniecki, Szymon Grabowski, and Wojciech Rozenberg. Image preprocessing for improving ocr accuracy. In *2007 International Conference on Perspective Technologies and Methods in MEMS Design*, pages 75–80, May 2007.

[2] Matteo Brisinello, Ratko Grbić, Matija Pul, and Tihomir Anđelić. Improving optical character recognition performance for low quality images. In *2017 International Symposium ELMAR*, pages 167–171, Sep. 2017.

[3] John Canny. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8(6):679–698, June 1986.

[4] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing (3rd Edition)*. Prentice-Hall, Inc., 2006.

[5] Wojciech Korobacz. Przetwarzanie wstępne zdjęć paragonów do celów rozpoznawania: praca magisterska. Master's thesis, Wydział Informatyki PB, 2017. (In Polish).

[6] Witold Malina and Maciej Smiatacz. *Cyfrowe przetwarzanie obrazów*. Akademicka Oficyna Wydawnicza EXIT, 2008. (In Polish).

[7] Nobuyuki Otsu. A Threshold Selection Method from Gray-level Histograms. *IEEE Transactions on Systems, Man and Cybernetics*, 9(1):62–66, 1979.

[8] William K. Pratt. *Digital Image Processing*. Addison-Wesley Publishing Company, 1991.

[9] John C. Russ and F. Brent Neal. *The Image Processing Handbook*. CRC Press, Inc., 2016.

[10] Khalid Saeed, Marek Tabedzki, Mariusz Rybnik, and Marcin Adamski. K3M: A universal algorithm for image skeletonization and a review of thinning techniques. *Applied Mathematics and Computer Science*, 20(2):317–335, 2010.

[11] Jaakko Sauvola and Matti Pietikäinen. Adaptive document image binarization. *Pattern Recognition*, 33(2):225–236, 2000.

[12] Linda G. Shapiro and George C Stockman. *Computer Vision*. Prentice Hall, 2001.

[13] Pooja Sharma and Shanu Sharma. Image processing based degraded camera captured document enhancement for improved ocr accuracy. In *2016 6th International Conference – Cloud System and Big Data Engineering (Confluence)*, pages 441–444, Jan 2016.

[14] Mande Shen and Hansheng Lei. Improving ocr performance with background image elimination. In *2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, pages 1566–1570, Aug 2015.

[15] Marek Tabedzki, Mariusz Rybnik, and Khalid Saeed. New results for view-based feature extraction method for handwritten words recognition without segmentation. In *1st International Conference on Image Processing & Communications, Poland*, pages 193–200, 2009.

[16] Marek Tabedzki, Khalid Saeed, and Adam Szczepanski. A modified K3M thinning algorithm. *Applied Mathematics and Computer Science*, 26(2):439–450, 2016.

[17] Øivind Due Trier and Anil K. Jain. Goal-directed evaluation of binarization methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(12):1191–1201, Dec 1995.

[18] Christopher Wiraatmaja, Kartika Gunadi, and Iwan Njoto Sandjaja. The application of deep convolutional denoising autoencoder for optical character recognition preprocessing. In *2017 International Conference on Soft Computing, Intelligent System and Information Technology (ICSIIT)*, pages 72–77, Sep. 2017.

PRZETWARZANIE WSTĘPNE ZDJĘĆ PARAGONÓW DO CELÓW ROZPOZNAWANIA

Streszczenie Tematem tej pracy są metody przetwarzania wstępnego obrazów, zastosowane do zdjęć przedstawiajacych paragony. Celem jest poprawa ich jakości, pozwalająca zwiększyć skuteczność działania oprogramowania do rozpoznawania tekstu. Autorzy mieli na uwadze głównie trudne przypadki – zdjęć robionych „z ręki", przy słabym oświetleniu. Praca opisuje przeanalizowane metody filtrowania, binaryzacji, wyszukiwania krawędzi, prostowania obrazu, oznaczania obszaru zainteresowania, ścieniania. Przedstawiono również wstępne wyniki testów z oprogramowaniem OCR na niewielkiej bazie obrazów. Przetwarzanie wstępne pozwoliło na poprawę identyfikacji znaków o 25%. W końcowej części przedstawiono wnioski oraz plany przyszłej pracy.

Słowa kluczowe: cyfrowe przetwarzanie obrazów, rozpoznawanie znaków

Artykuł zrealizowano w ramach pracy badawczej S/WI/2/2018.