Method of the optical recognition of technical documentation and the transformation of graphic information into machine-readable form for cognitive analysis

A A Dzyubanenko and A V Rabin
Saint-Petersburg State University of Aerospace Instrumentation (SUAI), ul. Bolshaya Morskaya, 67, lit. A, St. Petersburg, 190000, Russia

E-mail: aap@guap.ru, alexey.rabin@guap.ru

Abstract. The paper proposes the implementation of the method of optical recognition of technical documentation and the transformation of graphic information into a machine-readable form available for cognitive analysis, which is based on the methods of binarization and alignment of images, text segmentation and recognition. The use of the proposed method will provide a dramatic reduction in the costs of cataloging, checking the completeness and inventory of documentation, as well as an increase in design quality due to the semantic analysis of documentation using a knowledge base that is updated automatically. The article presents the development of the algorithm for optical recognition of a document, preparation of an image for optical recognition of a document, an example of the application of the Sauvola method for binarization of an image, and an analysis of the research results. The proposed implementation allows the text recognition on scanned / photographed documents.

1. Introduction
Optical character recognition (OCR) systems are designed for automatic input of printed documents into a computer [1].

These classes of algorithms have the following requirements for text recognition (International Phonetic Alphabet (IPA) principles):

- integrity;
- purposefulness;
- adaptability.

Consider the principles of IPA.

Integrity principle: an object is considered as a whole unit, consisting of connected parts. The connection of parts is expressed in spatial relations between them, and the parts themselves are interpreted only as part of the supposed whole, that is, within the framework of the hypothesis about the object. The advantage of a system following the above-described rules is expressed in the ability to more accurately classify a recognizable object, excluding from consideration many hypotheses at once that contradict at least one of the provisions of the principle.
Purposefulness principle: any interpretation of data has a specific purpose. Consequently, the recognition should be a process of making hypotheses about the whole object and purposefully testing them.

The principle of adaptability: the ability of a system to learn by itself. The information obtained during recognition is ordered, stored and subsequently used when solving similar problems.

The developed algorithm requires the implementation of each requirement [2].

The logic of document processing in the OCR system is determined by multilevel document analysis. Most modern OCR systems analyze a document in accordance with one of the following principles: "top down" or "bottom up".

The hierarchical scheme of such an analysis is presented below (figure 1).

Character recognition itself represents only one subtask within an OCR workflow, which usually consists of four main steps (figure 2), which can often be broken down into further subtasks.
1. Preprocessing: first, the input images must be prepared for a further processing. Typically, it includes a step that simplifies the presentation of the original color image by converting it to binary and sometimes to grayscale, allowing further image processing operations, as well as an alignment operation to get the pages in a vertical position, which simplifies subsequent steps. Additional procedures can be performed, such as de-warping to correct distorted scans, or smoothing, or stain removal to clean up scans. It can be helpful to crop the printable area ahead of time to remove unwanted scanning edges.

2. Segmentation: it is necessary to perform one or more segmentation steps, mainly depending on the available material and user requirements. After separating text areas from non-text areas, individual text lines must be identified and, depending on the recognition method used, further separated into individual characters. Optionally, non-textual elements can be further classified (images, ornaments, ...), while text areas can be broken down into more or less fine-grained semantic classes (running text, headings, marginalities, ...) already at the level layout. Another important sub-task is to determine the reading order, which determines the sequence of text elements (region and / or lines) on a page.

3. OCR: recognition of segmented lines (or single characters) results in a textual representation of printed input. Depending on the material at hand and the requirements of the user, this can be done either using existing models trained in different fonts that match the type used to some extent, so-called blended models, and / or using training models specifically designed for font recognition. on which he was trained.

4. Post-processing: the raw OCR output can be further enhanced in post-processing, for example by including dictionaries or language models. This step can be combined with manual correction, which usually occurs after automatic post-processing [3, 4].

2. Development of algorithm for optical recognition of document

The software for optical recognition of technical documentation and the transformation of graphic information into a machine-readable form available for cognitive analysis consists of separate modules, including their input-output ratios.

An example of the input data of the algorithm is presented in table 1.

| Name  | Description     | Format  |
|-------|-----------------|---------|
| Document | Scanned text document | pdf, jpg |

The output of the algorithm is presented in table 2.

| Name | Description     | Format |
|------|-----------------|--------|
| Text | Recognized text | txt    |

Figure 3 shows the steps in the workflow of the algorithm.

After obtaining scanned images and an additional stage of preparation, the original images can be placed in the work area. Image alignment and binarization are then applied to the scan, before the linear images required as input for character recognition are obtained in several steps such as region segmentation and extraction, as well as line segmentation. The character recognition output can either directly serve as the end result, or can be adjusted by the user, which allows training more accurate models, giving better recognition results [5].
Next comes the stage of text and image preprocessing, as well as preparing the image for optical recognition. In addition, the ratio of input and output of each module is always indicated with a description of the actual data with which each module operates and which is often created by combining information with a preprocessed grayscale or binary image [6].

At the main stage of preprocessing, the input images are prepared for further processing. An additional external preparation step can be performed prior to performing the two standard binarization and alignment subtasks.

3. Image preparation for optical document recognition

The input data of the image preparation unit are presented in table 3.

| Name  | Description                                                                 | Format  |
|-------|-----------------------------------------------------------------------------|---------|
| Image | Unfinished document image (image containing two scanned pages, page rotated in the wrong orientation, ...) | pdf, jpg |

The output data of the image preparation unit are presented in table 4.

| Name  | Description              | Format  |
|-------|--------------------------|---------|
| Image | Prepared image (one page vertically) | pdf, jpg |

The input image data will be in a vertical position and already segmented into separate pages, which can be easily achieved using ScanTailor (a computer program for processing images obtained
with a scanner) [7]. In addition, it is recommended that you remove excessive amounts of the scan background, although this is not required.

Figure 4 shows an example of valid and invalid inputs, which are also possible inputs and outputs of ScanTailor. In fact, ScanTailor is not a true submodule of the algorithm. However, it is still decided to list it as a module since this step is workflow related and the input images must still be added from external sources. It is possible to deal with unprepared images, such as those described in the input data, but this, of course, is not a recommended course of action [8].

Figure 4. Example of input and output of the stages of preparation and preprocessing of images.

In the preprocessing step, the distorted color image in the middle is converted to a flattened binary image on the right. Before aligning a color image, you must convert it to a binary form. The general scheme for converting a color image into binary is shown in figure 5 [9, 10].

![Diagram of image conversion process](image)

Figure 5. General scheme for converting a color image to binary.

Before binarization, the image must be converted from color to gray. To convert a color image to grayscale, use the following formula:

\[ Y_{x,y} = 0.299 \cdot R_{x,y} + 0.587 \cdot G_{x,y} + 0.144 \cdot B_{x,y}, \]
where $x, y$ are pixel coordinates, $Y_{x,y}$ is the pixel value in grayscale, $R, G$ and $B$ are the intensity of red, green and blue colors in a pixel.

To binarize the image, the local adaptive binarization method is used – the Sauvola method. The local binarization threshold is determined by passing the entire image by the window $w^*$. In the Sauvola binarization method the threshold $t(x, y)$ is determined by the following formula when used to calculate the mean and standard deviation $m(x, y)$ of the pixel intensity in the window $w^*$ around the pixel $(x, y)$:

$$t(x, y) = m(x, y) \left[ 1 + k \left( \frac{s(x,y)}{R} - 1 \right) \right].$$

In this formula, $R$ represents the maximum deviation ($R = 128$ for a grayscale image), and $k$ is a parameter that takes values in the range $[0.2, 0.5]$.

4. Application of the Sauvola method for image binarization

In the Sauvola method, there is the concept of an integral image. An integral image is an image whose pixel value determines the sum of all pixel values above and to the left of a position in the original image. Having calculated the integral image once, you can find the average value in the window by performing only three arithmetic operations instead of summing all the pixels in the window using the formula:

$$m(x, y) = \left( I \left( x + \frac{w}{2}, y + \frac{w}{2} \right) + I \left( x - \frac{w}{2}, y - \frac{w}{2} \right) \right) / w^2.$$

Next, the variance is calculated using the following formula:

$$s^2(x, y) = \frac{1}{w^2} \sum_{i=x-w/2}^{x+w/2} \sum_{j=y-w/2}^{y+w/2} g^2(i, j) - m^2(x, y).$$

In this formula, $g(i, j)$ is the pixel value at the point $(i, j)$.

The algorithm is a sequence of calculations using mathematical formulas implemented using standard calculation methods for high-level programming languages.

During the preprocessing stage, the input image is converted to the binary one. In addition, the alignment operation can be performed if the image has a "obstructed horizon". Blockage is a failure to observe the parallels between the horizon line in the image and the horizontal line of the monitor [11].

An example of the entry and exit of this step is shown in figure 6. While the image binarization step is a mandatory to facilitate upcoming imaging applications, page alignment is optional as its main purpose is to support a line segmentation process that works on segmented regions, which are in any case individually aligned in advance. However, depending on the degree of asymmetry in the original scans, it can be very useful to perform alignment already at the page level, as this can greatly simplify the segmentation of an area.

The final image preprocessing scheme is shown in figure 6.
Does the image have a blockage?

Yes

No

Image alignment

Converting an image to grayscale

Binarization with Sauvola algorithm

End

Binary image

Figure 6. Image preprocessing scheme.

For image alignment, the algorithm shown in figure 7 is used.

Binary image

Start

Applying RLSA to an Image

Finding the angle of inclination for each RLSA block

Calculating the mean value of an angle

Rotate the image

End

Aligned binary image

Figure 7. Image alignment algorithm.
In this paper, the algorithm detects skewing of text documents with high accuracy. The method depends on finding the horizontal image of the skewed document.

The average slope of the selected black connected components in the image is considered as the angle of inclination for the entire document, which is eventually rotated in the opposite direction by that amount to obtain the final corrected image. This method has advantages over others:

- this method is not limited, unlike methods using a projected histogram, to documents that have fairly small skews, which are usually less than ±10°;
- the problem of finding peaks in our method is absent in comparison with methods based on the Hough transformation;
- noise, character sub-parameters (dot on "i"), and interline links can reduce the accuracy of nearest neighbor clustering methods. There are no such problems in this method.

The study [12] proposed a block segmentation technique called RLSA ("run length smoothing algorithm"). The RLSA algorithm is applied to a binary sequence in which white pixels represent 0 and black 1. The algorithm converts the binary sequence \(X\) to the output sequence \(Y\) according to the following rules:

- 0 in \(x\) changes by 1 in \(y\) if the number of adjacent 0's is less than or equal to a certain limit \(p\);
- 1 in \(x\) has not changed in \(y\).

For example, if the sequence \(x\) is 00011000001100100001 and the value of \(r\) is 3, then the output sequence \(y\) is 11111000001111100001.

5. Analysis of software results
RLSA is applied to the image of a text document row by row, resulting in the bitmap shown in figure 8.

![Figure 8. Image of a text document using RLSA.](image-url)
• look for the topmost black point of the block, which lies at a distance $d$ to the left of the LRC, point C in figure 9;
• look for the lowest black point of the block, which lies at a distance $d$ to the left of the LRC, point D in figure 9;
• the middle point V is searched for from the two points found in steps 4 and 5;
• the angle of the straight line joining the two midpoints, U and V, is the tilt angle of the block, and therefore the tilt angle of the corresponding line of text.

For a word processing document, these steps are repeated for each selected block in the RLSA image. The average angle of all blocks is the skew angle of the entire document.

After conversion, we get an aligned binary image.

6. Conclusions
This paper proposes an implementation of the algorithm for optical recognition of technical documentation and transformation of graphic information into a machine-readable form available for cognitive analysis, which is based on the methods of binarization and alignment of images, text segmentation and recognition. The proposed implementation allows for text recognition on scanned / photographed documents.

The distinctive feature of the implementation of the algorithm is the presence of preliminary text segmentation in it, which improves the accuracy of optical recognition.

Acknowledgments
The paper was prepared with the financial support of the Ministry of science and higher education of the Russian Federation in the course of the applied research «The comprehensive project to create high-tech production of software tools for automatic analysis of documentation on paper and digital media using semantic-cognitive technologies for cataloging poorly structured information» (unique identifier of the project 075-11-2019-055, Decree of the Government of the Russian Federation N 218, 09.04.2010).

References
[1] Pervushin A 2012 The module of graphematic analysis in the system of processing Russian-language texts New information technologies in automated systems 15 187-90
[2] Rabin A V and Petrushevskaya A A 2020 Development of the algorithm for graphematic analysis and isolating of semantically significant constructions in poorly structured text Journal of Physics: Conference Series 1679(4) 042002
[3] Renu R S, Mocko G and Koneru A 2013 Use of big data and knowledge discovery to create data backbones for decision support systems Procedia Computer Science 20 446-53
[4] Chen D and Manning C D 2014 A fast and accurate dependency parser using neural networks Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing
[5] Rabin A V and Petrushevskaya A A 2020 Development of the formal model for the presentation of poorly structured and unstructured information *IOP Conference Series: Materials Science and Engineering* **862** 052076

[6] Rajpathak D, Chougule R and Bandyopadhyay P 2012 A domain-specific decision support system for knowledge discovery using association and text mining *Knowledge and Information Systems* **31** 405-32

[7] Rabin A V and Petrushevskaya A A 2020 Development of the formal model of the optical recognition mechanism for tabular documents *IOP Conference Series: Materials Science and Engineering* **862** 052078

[8] Litifi H, Kolski C, Ayed M B and Alimi A M 2013 A human-centred design approach for developing dynamic decision support system based on knowledge discovery in databases *Journal of Decision Systems* **22** 69-96

[9] Rabin A V and Petrushevskaya A A 2020 Hybridization of linguistic and statistical methods of text analysis for describing the subject area in the form of a fuzzy ontology *Journal of Physics: Conference Series* **1679**(4) 042008

[10] Rabin A V, Petrushevskaya A A and Sinitsin O V 2020 Methods and formal models of intelligent analysis of weakly structured data *IOP Conference Series: Materials Science and Engineering* **734** 012159

[11] *AstraVer: reliable and secure software* http://astraver.linuxtesting.org/review/

[12] Wong K Y, Casey R G and Wahl F M 1982 Document analysis system *IBM J. Res. Develop* **6** 647-56