Computer Engineering and Intelligent Systems
ISSN 2222-1719 (Paper) ISSN 2222-2863 (Online)
Vol.4, No.5, 2013

Finding Similarities between Structured Documents as a Crucial Stage for Generic Structured Document Classifier

Hamam Mokayed (Corresponding author)
Universiti Teknology Mara UiTM, Faculty of Information Technology and Quantitative Sciences
Universiti Teknologi MARA (UiTM) 40450 Shah Alam, Selangor Darul Ehsan, Malaysia
E-mail : Homammo@gmail.com
Azlinah Hj. Mohamed
Universiti Teknology Mara UiTM, Faculty of Information Technology and Quantitative Sciences
Universiti Teknologi MARA (UiTM) 40450 Shah Alam, Selangor Darul Ehsan, Malaysia
E-mail : Azlinah@tmsk.uitm.edu.my

Abstract
One of the addressed problems of classifying structured documents is the definition of a similarity measure that is applicable in real situations, where query documents are allowed to differ from the database templates. Furthermore, this approach might have rotated [1], noise corrupted [2], or manually edited form and documents as test sets using different schemes, making direct comparison crucial issue [3]. Another problem is huge amount of forms could be written in different languages, for example here in Malaysia forms could be written in Malay, Chinese, English, etc languages. In that case text recognition (like OCR) could not be applied in order to classify the requested documents taking into consideration that OCR is considered more easier and accurate rather than the layout detection.
Keywords: Feature Extraction, Document processing, Document Classification.

1. Introduction

In the light of globalization and its radical economic and technological evolution, offices and banks are forced to manage a large number of documents in different formats from customers, partners, etc. which must be classified and organized quickly in order to provide a differential service. This involves staff training costs, the establishing of procedures that ensure the categorization criteria, verification process, etc. which increase costs and document processing time, without ensuring the utmost quality in the process.
In this work we find a way to extract similarities between the documents based on reference lines and distinctive blobs of structured document. A dynamic tilting technique based on clustering has been proposed on the adaptive thresholded image. The tilted image is used as inputs to the two different feature extraction modules, the first one is based on extracting the reference lines (vertical, horizontal, and diagonal), and the second detects distinctive data in the structured document such as Logo and title area. The software is developed using C++ programming language and its block diagram is shown as in Fig. 1.

This paper has been organized with the introduction first. The pre-processing steps are described in the next section. The proposed tilting module is next described. Feature extraction modules are discussed in the sections that follow.

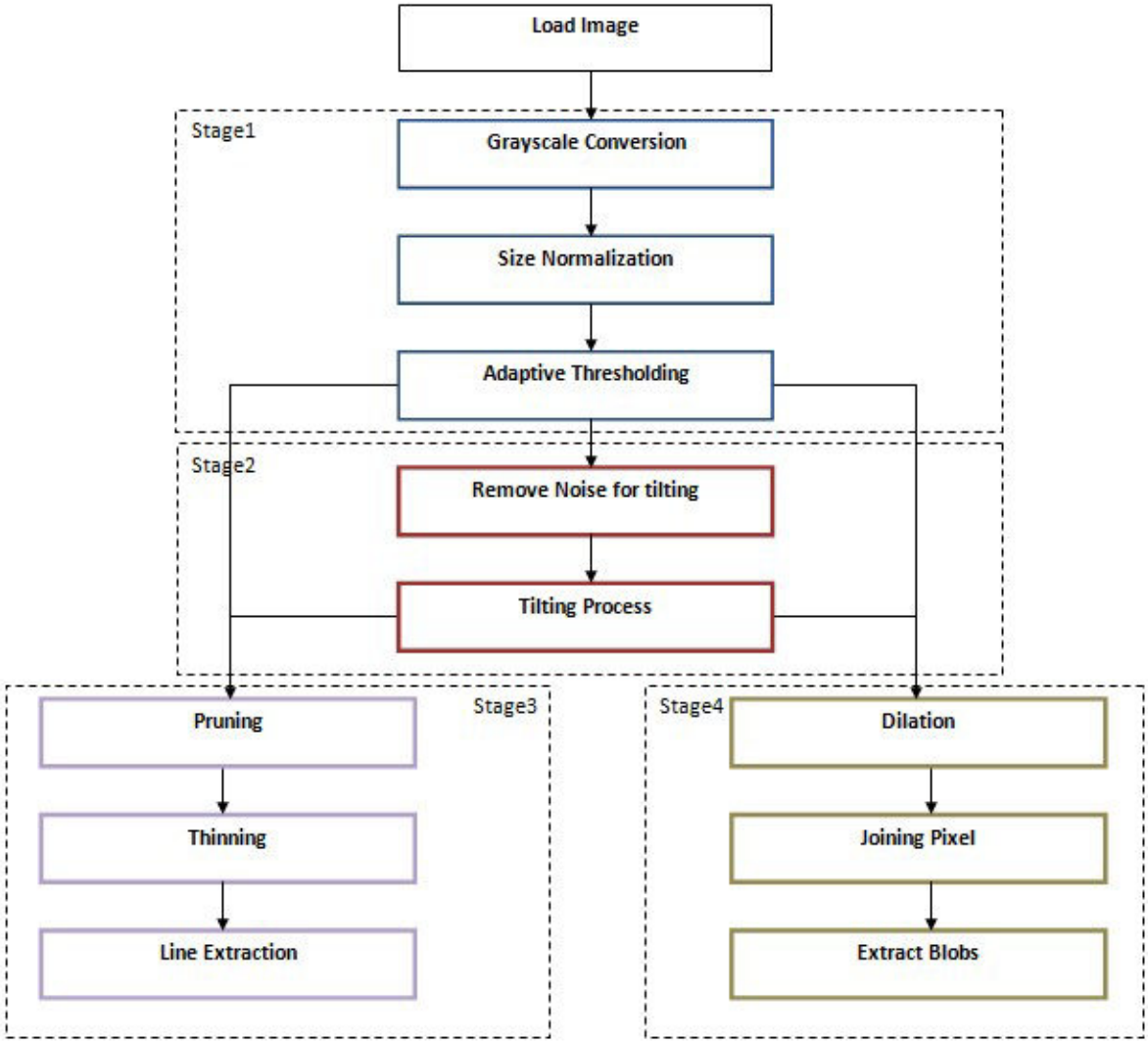


Figure 1. Block Diagram of the Proposed System.

2. Pre-processing Stage

The pre-processing part of the proposed system consists of three steps as is shown in stage1 in Fig.1

2.1 Grayscale Conversion

This method tries to convert the scanned image to grayscale

2.2 Size Normalization

The important idea underlying the stage carried out is to get normalized length images to enhance system performance by scaling each image both horizontally and vertically to get the same scaled image.

$$x_i = \frac{x_i^o - x_{\min}}{x_{\max} - x_{\min}} W$$

$$y_i = \frac{y_i^o - y_{\min}}{y_{\max} - y_{\min}} H$$

Where

(x_i^o, y_i^o) are the original points

(x_i, y_i) are the points after transformation.

$x_{\min} = \min_i \{x_i^o\}$

$x_{\max} = \max_i \{x_i^o\}$

$y_{\min} = \min_i \{y_i^o\}$

$y_{\max} = \max_i \{y_i^o\}$

W is the width of the normalized data = 500

H is the height of the normalized data = 500

Figure 2. Example of Normalization Process Applied on the Scanned Image.

2.3 Adaptive Thresholding

The system thresholding technique is developed based on the ordinal structure fuzzy logic model which has an advantage due to their ability in handling multiple inputs and outputs. Based on the 3 highest accurate pre-developed thresholding techniques such as SIS, Deravi, and Rosenfeld the fuzzy logic model is developed to predict the value of the pixel after thresholding whether it is black or white. Software has been developed for such purpose and the performance of the structured document classifier is compared to the different thresholding techniques. The results based on using SOFM for thresholding show a high level of accuracy which can be used

for many applications.

3. Tilting Structured Document

Tilting and skewing the scanned structured document is considered as a crucial stage as it has a great impact on the accuracy of the whole structured document classification system. Various techniques have been developed to calculate the value of proper rotation angle to adjust the tilted image. However, most of these techniques were applied as pre-stage of OCR systems and had the problem of expensive computational costs. In the proposed system, tilting will be accomplished by using the reference lines in the structured documents as a base of calculation. The objective is to find a way with lower computation time and higher accuracy results over different layouts that might contain non-text areas. Several algorithms were implemented to find the proper skew angle of tilted image [4,5], some of the applied methods can be classified as the following:

- Projection-based Methods [6,7]: basic concept of these methods is to use the extracted features out of projection profile calculation to determine the skew value.
- Hough transform-based methods [8,9]: methods depend on transforming the coordinates from Cartesian to polar and using the new values to calculate the skew angle.
- Cross – correlation methods [10]: vertical deviation along the image is used to calculate the skew of image.
- Clustering – based methods: these methods try to find a way to connect the pixels and use this connection to calculate the skew value. Shivakumara and Kumar [11] apply the nearest neighbor technique by labeling the connected black pixels and grouping them in blocks, these blocks were used to calculate the skew value. In the other hand, Sarfraz et al. [12] identify words within the same line to produce blob used in skew angle calculation. Chou et al [13] proposed a method based on drawing scan lines over the document at various angles. The skew angle is estimated by constructing parallelograms with these scan lines that cover objects in the image.

3.1 Clustering – Based Proposed Approach

As structured document contain of different components such as lines, text, check boxes and circles, logos, etc., lines are the essential layout structure of the document and can be detected more reliably than other components. So, we calculate the skew angle based on the document detected lines. Proposed approach is based on looking for the connected lines and estimating the optimum skew angles of the reference line. The idea behind this approach is that the distance between the consecutive points of the line is too small than the other components. So if we are able to detect the lines and find the reference line (line which has the largest number of pixels) out of them. We then estimate the skew angle based on that line.

To achieve the previous mentioned aim, the following stages should be executed:

3.2 Noise removal

After binarizing the image, this stage hopes to eliminate the randomly distributed noisy pixels that might affect on the calculation of the tilting value. The method is based on segmenting the whole binarized image to 5*5 sub-images, it statistically examines the intensity values of the active pixels in the sub window by counting total number of them and comparing the value with a threshold value in order to decide whether to flip their values to background or not. The size of the window has to be large enough to cover sufficient foreground and background pixels and to avoid either keeping the noise in case of too small size or disconnecting the line because of choosing too large region. After applying the proposed method to remove the noise, the values of the first and last three columns and rows of the image will be set to zero to avoid the edge noise caused by the low scanning resolution of the image.

$$N_a = \sum_{c+5}^{i=c} \sum_{j=r+5}^{j=r} P_{i,j}$$

If $N_a < \text{Th}$ then $\sum_{c+5}^{i=c} \sum_{j=r+5}^{j=r} P_{i,j} = 0$

Else Do Nothing
Where
N_a Total Number of active pixels within 5*5 window.
Th = (is* is + is) / 3 : is=2 increment Step for a pixel to constitute the sub window

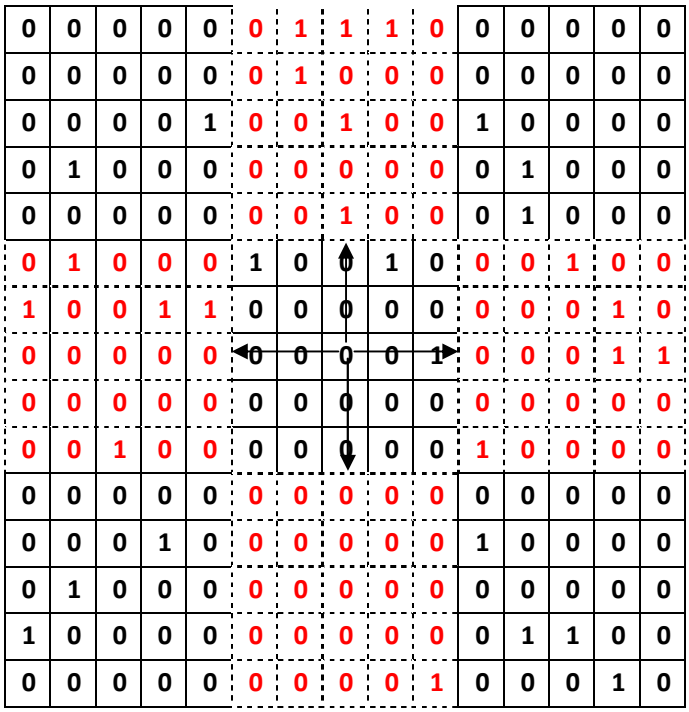
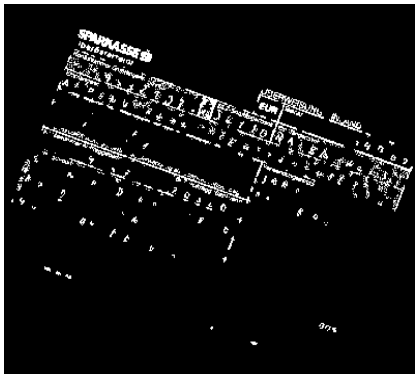


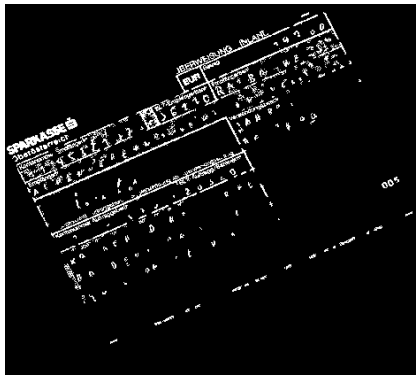
Figure 3. Noise removal process.

3.3 Tilting and skewing

Method using clustering technique will be presented to calculate the value of the angle. before starting the proposed method, a calculation of the first black pixel Pi(xi,yi) will be done and based on the condition of finding a small value of Y axis histogram at yi (histY[yi]), the angle calculation will start, otherwise no skewing is required .



Sample of image tilted to left



Sample of image tilted to Right

If xi < width/2 then image might be tilted to left, otherwise right

Figure 3. Examples of Different Tilted Images.

Four referencing points are used to calculate the best value of the angle as the following:

Pclk: first black pixel by scanning the columns form left to right

Pcrk: first black pixel by scanning the columns form right to left

Prtk: first black pixel by scanning the rows from top to bottom

Prbk: first black pixel by scanning the rows form bottom to top

A sequence of operations should be executed over each of the previous mentioned referencing points to calculate the most accurate tilting angle.

Applying the proposed method over Pclk as an example

$$|x_{cli} - x_{cli+1}| < x_{climin} \text{ and } |y_{cli} - y_{cli+1}| < y_{climin} \text{ then } P_{cli}, P_{cli+1} \in G_{cl}$$

$$: x_{climin} = 3, y_{climin} = 20, i \in [(y_{clk} * width + x_{clk}), (width * height)]$$

$$\forall P_{cli}, P_{cli+1} \in G_{cl}, xdiff_{cl}[i] = |x_{cli} - x_{cli+1}|$$

$$xdiff_{max\_cl} = max_i\{xdiff_{cl}[i]\}$$

$$\forall P_{cli} \in G_{cl} \text{ and } (xdiff_{cl}[i] = xdiff_{max\_cl}) \text{ then } P_{cli} \in SG_{cl}$$

$$\forall P_{cli} \in SG_{cl} \text{ and } (|y_{cli} - y_{cli+1}| > 5) \text{then } y_{cli} = -1$$

$$\{\text{while } (P_{cli+k} \in SG_{cl} \text{ and } y_{cli} <> -1)$$

$$(Num_{cl}[j] = Num_{cl}[j] + 1) \text{ and } (k = k + 1) \text{ and } (P_{cli+k} \in SG_{clj})\}$$

After calculating Number of pixels in each group of the referencing points (Num []), group which has the highest number of pixels will be used to calculate the skewing angle. Let us suppose that SG_{rt2} is the nominated group

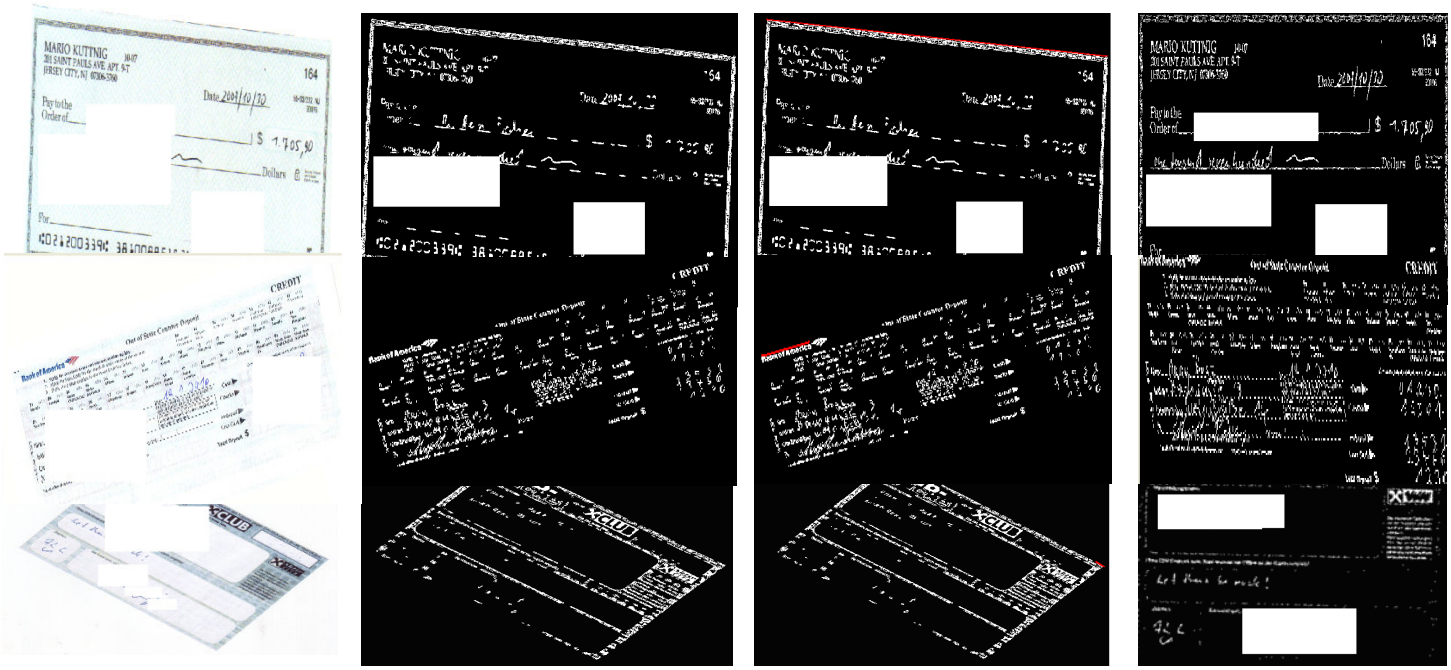
$$xref_{min} = min_i\{x_i\} : x_i \in (x_{clk}: SG_{rt2})$$

$$yref_{min} = min_i\{y_i\} : y_i \in (y_{clk}: SG_{rt2})$$

$$xref_{max} = max_i\{x_i\} : x_i \in (x_{clk}: SG_{rt2})$$

$$yref_{max} = max_i\{y_i\} : y_i \in (y_{clk}: SG_{rt2})$$

$$Angle = \tan^{-1}\frac{(yref_{max} - yref_{min})}{(xref_{max} - xref_{min})}$$



1- Original Image 2-thresholding and removing noise 3- Reference line used for tilting 4- Tilted Image

Figure 5. Result of Applying Tilting Process on Different Samples.

4. Feature Extraction

Finding similarities in document has offered great challenge to the solution provider due to the huge amount and unsorted documents which were collected every day and differed in terms of outlook, layout, size, meaning etc [14]. In addition to, finding generic features that able to interpret different types of forms is not feasible using present techniques, it is possible to develop a specific method to process the specific type of forms contained in the documents. A little research had concentrated on the layout of document, as a primary key for the feature extraction to be used in classification. The most recent work applied watermark embedding method to obtain layout information of the document with concentration of documents integrity detection and offered secret information to be hidden, and it was robust to hard and soft documents [15].

As the proposed technique is language separable, reference lines and blobs could be considered as good features in order to classify the structured documents. Next section will clarify the way of extracting features and all the pre-processing stages that carried out to enhance the extraction process.

4.1 Extraction of reference lines

Proposed system recognizes ruled lines by extracting vertical, horizontal, and diagonal lines from the binary image, obtained by adaptive thresholding method. Proposed line extraction algorithm uses pruning as one of the morphological operations, thinning, connecting gaps of broken and disconnected lines, and the clustering technique to extract the reference lines and use them as a one of the feature in the classification system.

a. Noise removal

After thresholding and skewing the image, this stage hopes to eliminate the noisy pixels and spurs on the end points in order to enhance the lines extraction process. Pruning as one of the morphological operations is applied to remove these noisy scattered pixels which are not related to any one of the objects in the document.

The applied pruning uses two basic structuring elements and iterates them to produce other six structuring elements. All structuring elements are applied once over the processed image to get the required results.

0	0	0		0	0	0		0	0	0		-1	0	0		-1	-1	0		0	-1	-1
0	1	0		0	1	0		-1	1	0		-1	1	0		0	1	0		0	1	0
0	-1	-1		-1	-1	0		-1	0	0		0	0	0		0	0	0		0	0	0
1st structured Element				2nd structured Element				1st clockwise(1)				2nd clockwise(1)				1st clockwise(2)				2nd clockwise(2)		

0	0	-1		0	0	0	
0	1	-1		0	1	-1	
0	0	0		0	0	-1	
1st clockwise(3)				2nd clockwise(3)			

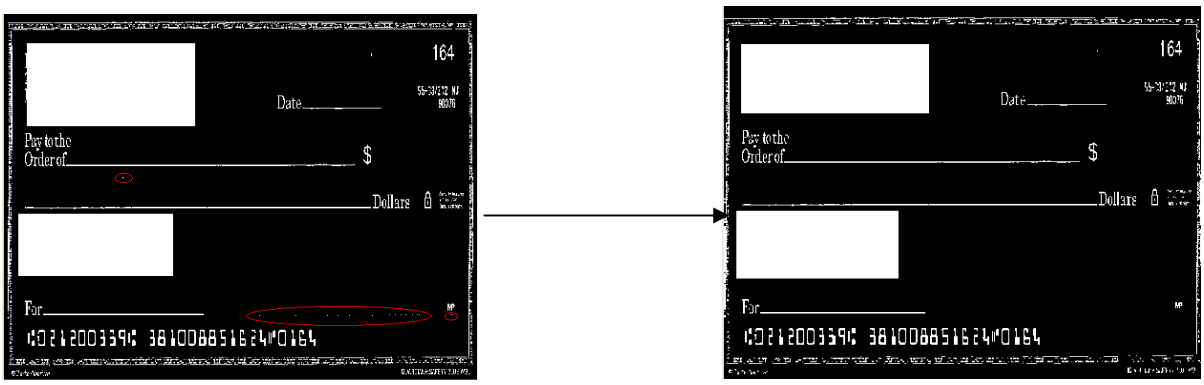


Figure 6. The result of Applying Pruning Structuring Element ver Structured Document.

b. Thinning

Thinning is usually used in applications that aiming to extract lines for different purposes such as OCR to improve the recognition accuracy [16]. Thinning is applied to reduce thickness of each line to just a single pixel line in order to facilitate the extraction process of the reference lines. The applied thinning method uses two basic structuring elements and iterates them till the convergence. Iterating the elements will produce eight structuring elements used for skeletonizing the whole image. As shown in the below figure, the image is first thinned by two basic structuring elements, and then with the remaining six 90° rotations of the two elements. The process is repeated till there is no change by applying anyone of the structuring element.

0	0	0		-1	0	0		1	-1	0		-1	1	-1		1	1	1		-1	1	-1
-1	1	-1		1	1	0		1	1	0		1	1	0		-1	1	-1		0	1	1
1	1	1		-1	1	-1		1	-1	0		-1	0	0		0	0	0		0	0	-1
1st structured Element				2nd structured Element				1st clockwise(1)				2nd clockwise(1)				1st clockwise(2)				2nd clockwise(2)		

0	-1	1		0	0	-1
0	1	1		0	1	1
0	-1	1		-1	1	-1
1st clockwise(3)				2nd clockwise(3)		

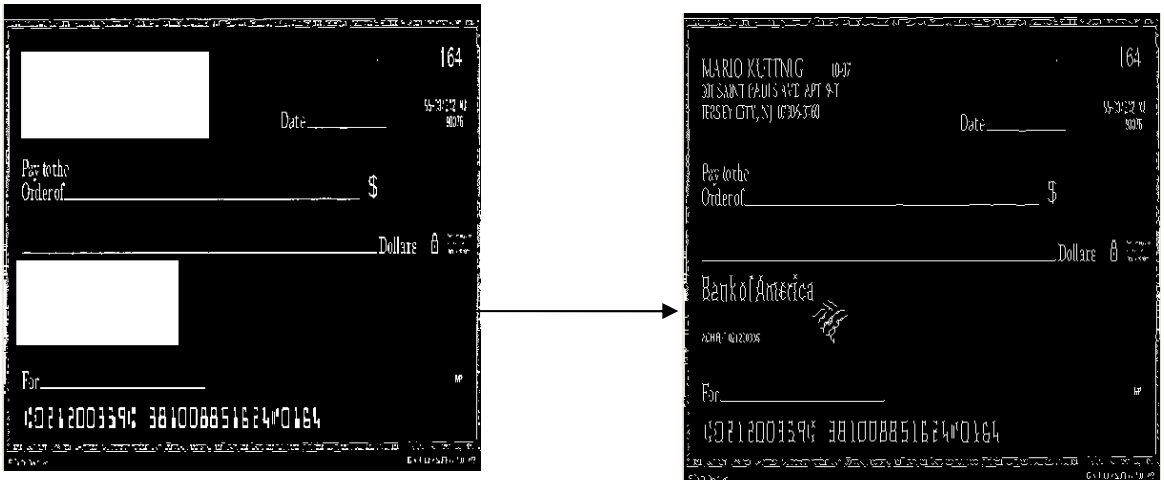


Figure 7. The Result of Applying Thinning Structuring Element over Structured Document.

c. Joining broken lines

before starting the proposed method to detect the lines, values of the histograms (histY , hsitX) will be used as inputs to decide whether to process joining for horizontal and vertical lines or not.

$$\forall i \in [0, Height] \text{ and } (hist_X[i] > th_{length}) \text{ then do the processing : } th_{length} = 50$$

$$\forall j \in [0, width] \text{ and } (hist_y[j] > th_{length}) \text{ then do the processing : } th_{length} = 50$$

After specifying the target column or row, the decision of connecting tow groups of sequential active points separated by spaces will be depended on the number of active pixel in the first part, number of non active points (Numspaces), and number of active points (Num2active) in the second part. As shown in the following formula

$$\text{if}\left\{\left(\frac{\text{Num}_{spaces}}{(\text{Num}_{spaces}+\text{Num1}_{active}+\text{Num2}_{active})}<0.1\right)\text{and}\left(\frac{\text{Num1}_{active}}{(\text{Num}_{spaces}+\text{Num1}_{active}+\text{Num2}_{active})}>0.2\right)\text{and}\left(\frac{\text{Num2}_{active}}{(\text{Num}_{spaces}+\text{Num1}_{active}+\text{Num2}_{active})}>0.2\right)\right\}$$

then connecting otherwise do nothing

d. Line Segments Detecting and Grouping

Proposed method will be used to extract the vertical, horizontal, and diagonal lines out of the previous processed binary image. First of all, pruning operation is applied to find the end points of the lines and use them as starting points of the line tracing process. The applied pruning method is exactly like the once implemented before but it will be used here to mark the pixels which are detected by the process not to eliminate them. After specifying the pixels, the following steps will be executed

- Get one of the pruning points
- Mark the point in order not to be used anymore, save it as one of the line's pixel, increase the number of line pixels, and compare the coordinates to get the value of Xmin , Ymin, Xmax, Ymax of that line.
- Scan all the eight neighboring pixels and count how many pixels are active and not marked.
- In case the value retrieving back from stage 3 is
    - 1: repeat steps starting from 2
    - >1 obtain the coordinates of each active pixel, push them into a branch vector, and jump to step 5
    - 0: jump to step 5
- Analyze the detected line and decide whether it is one of the reference line or not by applying the following steps:
  Xdiff = Xmax - Xmin
  Ydiff = Ymax - Ymin
  Rval = (Ydiff / Xdiff) in case Ydiff < Xdiff , otherwise Rval = (Xdiff / Ydiff)
    - If (number of line's pixel > 30 and Rval > 0.2 and (Ydiff < 25 || Xdiff < 25)) then the line isn't one of the reference line and no need to save it; otherwise save the line
    - If the branch vector has any values get one of the value and jump to 2
- Repeat the steps starting from one.

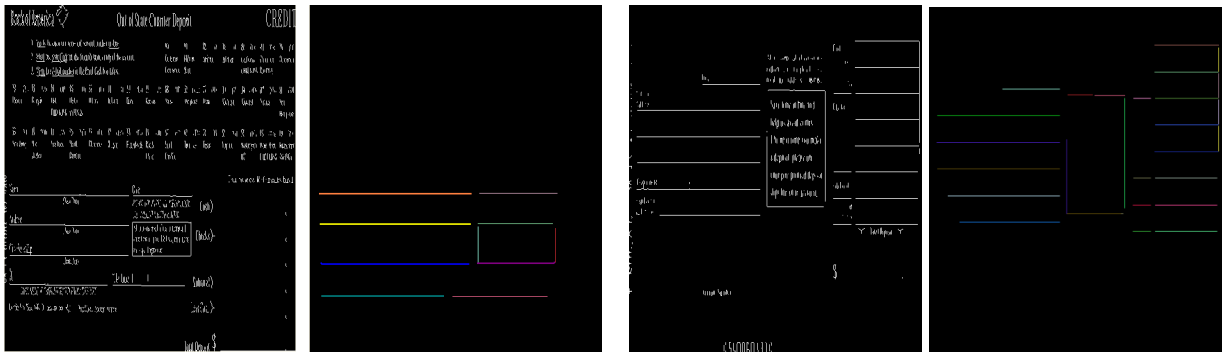


Figure 8. Applying Line Extraction Process over Two Different Samples.

4.2 Extraction of blobs

This stage tries to detect distinctive data in the structured document such as Logo and title area, the places of required information can be notices as connected regions in binary image. The aim of this stage is to enhance the accuracy by adding the useful information of the detected places as inputs to the classifier beside the information

extracted from the reference lines in the previous stage. Proposed blob extraction algorithm uses dilation as one of the morphological operations, joining pixel, and proposed technique to extract the blobs and use them as a one of the feature in the classification system.

a. Dilation

Dilation is usually applied to merge nearby regions by expanding all the white parts (active pixels) and facilitating the extraction process of the blobs. The applied method uses one structuring element (3*7) as shown in the figure below. The process is flipped all the neighboring pixels of the target pixel to one in case the pixel is active, otherwise nothing will change.

b. Joining Pixels

Before starting the proposed method to detect the accurate blobs, this step joins the connected target pixel into different blobs and removes non distinctive blobs that contain area below specific threshold.

$$\text{If Size of Blob} < Th \text{ then } Remove \text{ that blob}$$

$$\text{Size of blob} = \text{number of the active pixels inside the blob, } th = 800$$

-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1

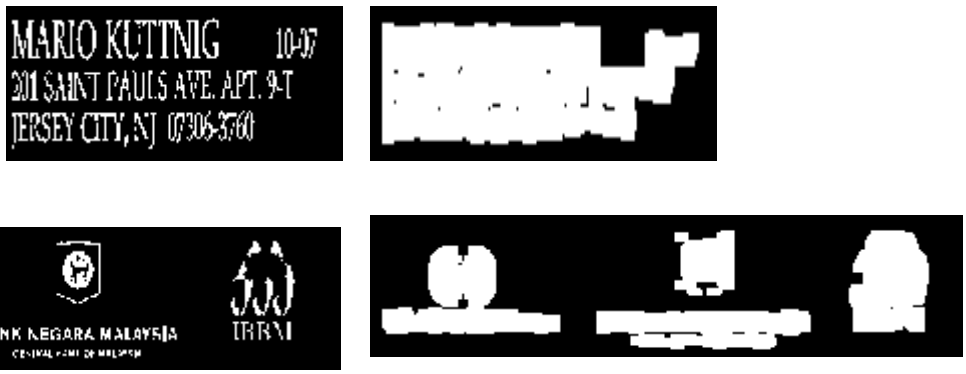


Figure 9. Applying Joining Pixels Process over Two Different Headers.

c. Determining distinctive Blobs

Due to the fact that not all the detected blobs could be used as unique features for the classifiers system, blobs should be achieving the following conditions in order to be nominated to the next stages.

First condition related to the dimension Ratio

$$\frac{Blob_{width}}{Blob_{height}} > DimensionRatio$$

Second condition is ratio of the height

$$\left(\frac{\text{Blob}_{\text{height}}}{\text{image}_{\text{width}}} > HieghtRatio1\right) \text{and} \left(\frac{\text{Blob}_{\text{height}}}{\text{image}_{\text{width}}} < HieghtRatio2\right)$$

Third condition is ratio of the width

$$\frac{\text{Blob}_{\text{width}}}{\text{image}_{\text{width}}} * 100 < WidthRatio$$

Last condition is location of the blobs should be located at top 10% part of the image

$$\left(\text{Blob}_{y} + \text{Blob}_{\text{height}}\right) < 0.1 * \text{image}_{\text{height}}$$

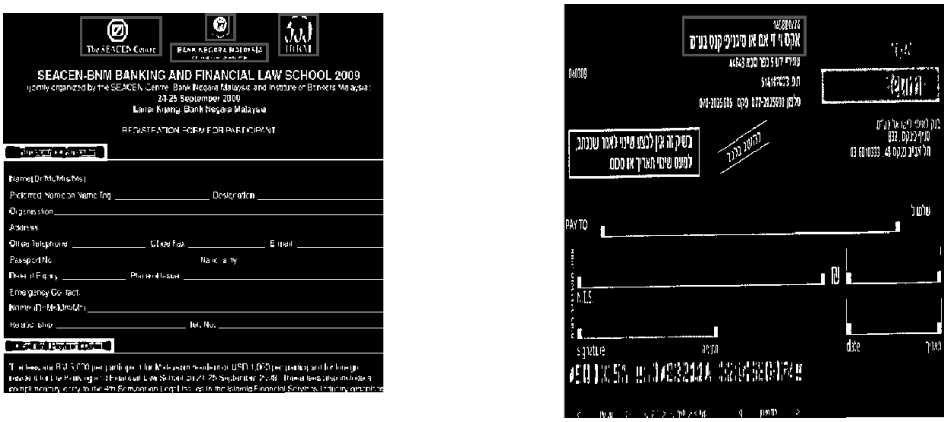


Figure 10. Applying Blob Extraction Process over Two Different Samples.

1- Conclusion

As the performance of the document classification depends upon the uniqueness of extracted features, the paper proposed to find both referencing lines and blobs to address the problem of retrieving similarities in document which has offered great challenge to the solution provider due to the huge amount and unsorted documents which were collected every day. Another potential avenue is that the proposed technique able to classify structured documents written in different language as it is mainly based on the layout of the form regardless the language used.

References

Zhang Junyou. (2010). A Quickly Skew Correction Algorithm of Bill Image. Paper presented at Third International Conference on Information and Computing

Daniel Lopresti & Ergina Kavallieratou. (2010) , Ruling Line Removal in Handwritten Page Images. Paper presented at International Conference on Pattern Recognition.

Hernâni Gonçalves, José Alberto Gonçalves, and Luís Corte-Real. (2011). A Method for Automatic Image Registration Through Histogram-Based Image Segmentation. IEEE transaction in image processing, VOL. 20, NO. 3.

Jonathan J. H (1998), Document Image Skew Detection: Survey and Annotated Bibliography, World Scientific, 40-64.

Hull, J., (1998). Document image skew detection: Survey and annotated bibliography. Document Analysis Systems II. World Scientific Pub. Co. Inc. pp. 40–64.

Postl, W., (1986). Detection of linear oblique structures and skew scan in digitized documents. In: Proc. 8th Internat. Conf. on Pattern Recognition, Paris, France, 687–689.

Li, S., Shen, Q., Sun, J., (2007). Skew detection using wavelet decomposition and projection profile analysis. Pattern Recognition Lett. 28 (5), 555–562.

Amin, A.; Fisher, S (2000). A Document Skew Detection Method Using the Hough Transform, Pattern Analysis

& Applications 3 (3) pp. 243-253.
Singh, C., Bhatia, N., Kaur, A., (2008). Hough transform based fast skew detection and accurate skew correction methods. Pattern Recognition 41 (12), 3528–3546.
Avanindra, S., (1997). Robust detection of skew in document images. IEEE Trans. Image Process. 6 (2), 344–349.
Shivakumara, P., Kumar, G.,(2006). A novel boundary growing approach for accurate skew estimation of binary document images. Pattern Recognition Lett. 27 (7), 791–801.
Muhammad Sarfraz , Zeehasham Rasheed (2008). Skew Estimation and Correction of Text using Bounding Box" Proc. of Fifth International Conference on Computer Graphics, Imaging and Visualization.IEEE CGIV 2008 pp259-264.
Chou, C.; Chu, S.; Chang, F (2007). Estimation of skew angles for scanned documents based on piecewise covering by parallelograms", Pattern Recognition 40 (2) . pp. 443-455.
Y. Cao, S.H. & Wang, H. Li (2002). Automatic recognition of tables in construction tender documents. Paper presented at Automation in Construction, 11 (5) ,573 – 584.
He, Y., Luo, L., Su, M., Shao, L., & Xiang, Z. (2011). Embedding and detecting watermarks based on embedded positions in document layout: Google Patents.
V.N Manjunath Aradhya (2007). Skew Estimation Technique for Binary Document Images based on Thinning and Moments, Journal of Engineering Letters, Vol 14, No.1, pp. 127-134.

This academic article was published by The International Institute for Science, Technology and Education (IISTE). The IISTE is a pioneer in the Open Access Publishing service based in the U.S. and Europe. The aim of the institute is Accelerating Global Knowledge Sharing.

More information about the publisher can be found in the IISTE's homepage: http://www.iiste.org

CALL FOR PAPERS

The IISTE is currently hosting more than 30 peer-reviewed academic journals and collaborating with academic institutions around the world. There's no deadline for submission. **Prospective authors of IISTE journals can find the submission instruction on the following page:** http://www.iiste.org/Journals/

The IISTE editorial team promises to the review and publish all the qualified submissions in a **fast** manner. All the journals articles are available online to the readers all over the world without financial, legal, or technical barriers other than those inseparable from gaining access to the internet itself. Printed version of the journals is also available upon request of readers and authors.

IISTE Knowledge Sharing Partners

EBSCO, Index Copernicus, Ulrich's Periodicals Directory, JournalTOCS, PKP Open Archives Harvester, Bielefeld Academic Search Engine, Elektronische Zeitschriftenbibliothek EZB, Open J-Gate, OCLC WorldCat, Universe Digtial Library , NewJour, Google Scholar