A New method for Displayed Mathematical Expression Detection Based on FFT and SVM

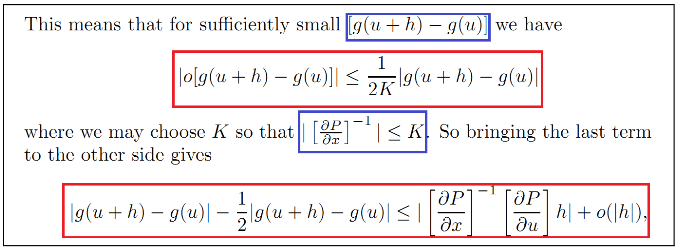
Bui Hai Phong*, Thang Manh Hoang †, Thi-Lan Le*
*MICA International Research Institute (HUST - CNRS/UMI2954 - Grenoble INP)
Hanoi Univeristy of Science and Technology, Hanoi, Vietnam
Email: thi-lan.le@mica.edu.vn; hai-phong.bui@mica.edu.vn
†School of Eletronics and Communications
Hanoi Univeristy of Science and Technology, Hanoi, Vietnam
Email: thang@ieee.org

***Abstract*—Extracting and recognizing mathematical expressions of scientific documents are key steps in the process of mathematical retrieval system, where the documents contain different components such as text, tables, figures, and mathematical expressions. There are several methods proposed to handle the components of documents. Those methods have investigated the feature of components based on the segmented text lines. This paper proposes a new method to detect mathematical presentation in the form of displayed expression from scientific documents using Fast Fourier Transformation. The test for the proposed method shows that the accuracy is from 94% to 98% for recognizing mathematical expressions on the benchmark datasets. It demonstrates the effectiveness of the proposed method in compared with the other ones.**

***Keywords*—Document analysis, Mathematical expression extraction, FFT, SVM**

I. Introduction

Mathematical notations have been commonly used since hundred years ago. In contemporary scientific documents, mathematical expressions are essential and indispensable. Recent years, number of scientific documents increase extraordinarily. These are valuable resources for research and education, especially for *Science, Technology, Engineering, Math* (STEM) disciplines. However, in order to efficiently access the information, scientific documents have to be digitized and converted into electronic format. In practical, typical scientific documents may contain various parts such as text paragraphs, tables, figures, mathematical notations and equations. Several *Optical Character Recognition* (OCR) systems (e.g. [1], [2], [3], [4]) have been produced to convert image-based contents to searchable data. Particularly, the work in [5] integrates additional mathematical notation recognition module into the system. This can improve recognition accuracy, however complicated computation is required. The existing systems perform accurately on text-only documents (e.g. [6], [7]). However, such those systems have appeared poor performance for mathematical expressions contained in heterogeneous documents. In this case, mathematical expressions are normally misrecognized or rejected. Therefore, in order to enhance recognition accuracy and apply existing OCR


This means that for sufficiently small $\boxed{g(u+h)-g(u)}$ we have

$$\boxed{|o[g(u+h)-g(u)]| \le \frac{1}{2K}|g(u+h)-g(u)|}$$

where we may choose K so that $\boxed{|\left[\frac{\partial P}{\partial x}\right]^{-1}| \le K}$. So bringing the last term to the other side gives

$$\boxed{|g(u+h)-g(u)| - \frac{1}{2}|g(u+h)-g(u)| \le |\left[\frac{\partial P}{\partial x}\right]^{-1}\left[\frac{\partial P}{\partial u}\right] h| + o(|h|).}$$

Fig. 1. Examples of displayed formula and inline formula in scientific document of Havard dataset

systems for scientific documents, mathematical expressions have to be detected and extracted efficiently.

Based on the appearance of mathematical expressions relatively to text paragraphs in the documents, they are classified into displayed expression and inline expression. A displayed expression (also, called isolated expression) is isolated from components while an inline expression (also, called embedded expression) is mixed with other components (normally text). Displayed expression and inline expression are marked in red and blue respectively in Fig. 1. A number of researches are proposed to classify displayed and inline expressions in text paragraphs. In general, there are two phases in the process of classification (e.g. [5], [8], [9]). Firstly, document layout is segmented into line by line based on the connected components or using existing OCR systems. Then, displayed expressions are classified from ordinary text lines by using various feature extraction techniques.

In this paper, a new method for detection of displayed expression is presented. The process is that the image of document layout is analyzed, and ordinary text lines and displayed expressions are extracted using OCRopus [2] - a popular open source software. After that, displayed expression is evaluated using the combination of *Fast Fourier Transformation* (FFT) and *Mean Square Error* (MSE) [10] to extract features. Finally, threshold and *Support Vector Machine* (SVM) [11] are used to classify. To our best knowledge, this is the first time that FFT is utilized for this task. The usage of other Wavelet Transform of image is not included in the scope of work. By comparing with previous works such as [8] and [9], it indicates that the proposed method allows to achieve better accuracy.

The rest of the paper is organized as follows. Section II describes the benmark datasets. Section III overviews significant related works. Section IV presents the detail of the proposed method. In Section V, experimental results are shown and discussed. Finally, Section VI gives the conclusion and the future work.

II. Dataset Description

In this work, we utilize two benchmark datasets: Harvard Mathematical Textbooks Dataset [12] and InftyCDB-2 [13]. The Harvard dataset contains some mathematical textbooks authored by Shlomo Sternberg at Harvard University. Those books in the PDF format are converted into images at the resolution of 3500×5700 for OCRopus requirements. These are binary images. Besides, the InftyCDB-2 is synthetic dataset of Infty Project [13] that is a commercial OCR system. It contains thousands of English, French, German articles.

It is noted that Harvard dataset contains documents that are well edited in scientific styles, therefore their components including mathematical expressions and text are well formatted. In contrast, Infty dataset is provided for testing OCR algorithms, its format is not formal, character styles change somewhere.

III. Related Works

The earliest work presented in [14] is the system which segments and extracts mathematical expressions from printed documents. Input document image is scanned line-by-line to extract adjacent black pixels. Then, pixels are considered to form small segments. Neighboring segments are merged into text lines if white space between them is smaller than a certain threshold. Text lines are then labeled text and displayed expression based on a feature extraction and Bayes decision rules. Two observed features are used to decide a line to be a text line or a displayed expression, i.e. the average height and the average of the white space between two lines. Testing experiments were carried out with their private data and the performance was shown without any comparison.

The method reported in [5] employed two more additional features to enhance the accuracy of mathematical expression recognition, i.e. the variation of characters' position and the statistical occurrence of mathematical symbols (e.g. '+', '-', '(', ')', '{', '}'). The threshold was set to classify displayed mathematical expression and ordinary text lines. Testing results were obtained using benchmark datasets including a part of Infty dataset. The highest accuracy of 97,2% was claimed for displayed expressions. However, it is noted that two commercial OCR systems were integrated into this work.

Recently, in the work of W.Chu and F.Liu [9] the method presented in [5] was developed by adding a novel feature, i.e. variation of characters' centroid, and the classification was carried out using SVM. The technique for text line segmentation was based on sign detection and text localization in natural images. The accuracy of 95% for displayed expression detection was reported.

In [8], the method embeds mathematical expression recognition into OCRopus [2] system that has been considered a powerful tool, easy to use and extend. After obtaining text lines using one CP (Column Projection) algorithm supported by OCRopus, this method is an extension of the work presented in [5] to identify mathematical expression.

In this paper, five features are considered to classify displayed expressions from ordinary text lines; those are:

(1) the proportion of white space of above and below text lines and white space of two consecutive text lines.

(2) variation of lowest position of characters in text lines.

(3) the proportion of height of each text line and average height of all text lines

(4) ratio of width and height of characters in text lines.

(5) left indent of text lines.

The vector of five features is used for classifying mathematical expressions and text lines with the help of SVM. Although this work computes sophisticated features, the efficiency is highly dependent on testing datasets. Actually, for some scientific document datasets like Harvard [12], these features are proper. In contrast, for synthetic datasets like Infty [13], extracted features are not really useful.

Recently, *Convolutional Neural Network* (CNN) has emerged as a powerful technology for object recognition and particularly for mathematical expression recognition. By using this framework, multitasks in recognition are processed simultaneously, human effort for hand-crafted feature extraction is reduced. The work [15] adopts sliding window based CNN framework and obtains accuracy of 87% for different mathematical symbols recognition. In this research, isolated mathematical expression images are captured by mobile phone. Therefore segmentation of mathematical symbols is not as complicated as in document. Context information of captured images is employed for locating and recognizing mathematical symbols. The limitation is that it is not able to perform structural analysis step to completely interpret mathematical expression to string.

In this work, we focus on a new approach to extract displayed mathematical expressions in scientific document. Similar to [8], document layout analysis and text line extraction are performed by using text line finding technique supported by OCRopus software. The main contribution of this paper is in classification process. We investigate and properly apply Fast Fourier Transformation technique to obtain image transformation values. After analytically comparing these values, classifiers such as threshold and SVM [11] are used for the classification.

IV. Proposed Method

The proposed method consists of two phases as presented in Fig. 2. The first and second phases are marked in blue and red, respectively. In the first phase, document images are preprocessed to remove noises in order to reduce the ambiguity. Input images are grey and then the images are converted

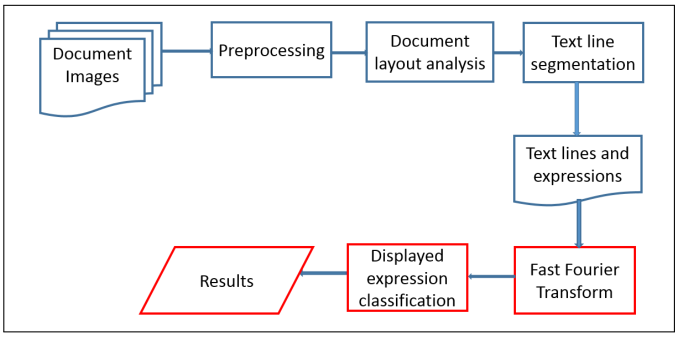


Fig. 2. Flowchart of the proposed system

to binary. In this phase, document layout is also analyzed to separate ordinary text line and displayed expression from other components, using OCRopus. In the second phase, displayed expression is classified from ordinary text line.

Algorithm 1 Algorithm of Displayed Mathematical Formula Detection

procedure DISPLAYED FORMULA DETECTION
    Input: Document Images
    Output: Classification Result
  preprocess(Documents) ;
  lines $\leftarrow$ *extractlines* ;
  V1= $\emptyset$, V2= $\emptyset$, $V3 = \emptyset$;
  for *line(i)* in *displayed formula training images* **do**
    for *line(j)* in *displayed formula training images* **do**
      $V1 \leftarrow$ *MSE(Fscore(i),Fscore(j))* ;
    end for
  end for
  for *line(i)* in *displayed formula training images* **do**
    for *line(j)* in *common text training images* **do**
      $V2 \leftarrow$ *MSE(Fscore(i),Fscore(j))* ;
    end for
  end for
  for *line(i)* in *displayed formula training images* **do**
    for *line(j)* in *test images* **do**
      $V3 \leftarrow$ *MSE(Fscore(i),Fscore(j))* ;
    end for
  end for
  svmtrain(V1,V2) ;
  svmclassify(V3) ;
  Return *Classification Result*
end procedure

After the first phase, a collection of lines that are ordinary text lines and displayed expressions are obtained. In the second phase, in order to classify displayed expression from text lines, the FFT is applied to the lines. The phase and magnitude values of FFT of every line are combined to formulate overall evaluation scores. These scores are considered using *Mean Square Error* (MSE) to clarify displayed expression from ordinary text. Actually, evaluated score of ordinary text line is much higher than displayed expression. Based on scores, threshold and SVM are invoked for the binary classification. Principle techniques are described in next parts.

A. *Phase 1 - Text line extraction*

A document image may contain noises, Gaussian, isolate noises, border noises, marginal noises. For example, some noises can be misrecognized as dot ('.') or minus ('–'), characters. The noise is determined by the size of pixel group and noise removal is carried out using local thresholding method. After that, the document image is converted into binary image for using next step. Then, binary image is analyzed to separate text lines and displayed expressions from figures, tables. Actually, input binary image is divided into candidate regions. Then, features are extracted for each candidate region. Based on these features, each component is classified by using logistic regression. There are several software packages developed for these processing steps. In this work, OCRopus open source software is adopted due that it is a powerful, easy-to-use tool. The software provides three principle modules:

(1) Document layout analysis: input document image is analyzed to get text regions, paragraphs, reading order.

(2) Text line recognition: each text line is separated based on connected components and white space computation. It should be noted that ordinary text line is recognized with high accuracy, sophisticated mathematical expressions are sometimes recognized with errors. Errors are filtered to suitable for the system.

(3) Statistical language modeling: the module process languages of input documents. Multiple languages such as English, French, German are supported for recognition.

A script file containing OCRopus commands is created to extract line-by-line from document images. The image resolution is set at least 3500×5700. A collection of text lines and displayed expressions are obtained. An example of the extraction is shown in Fig. 3.

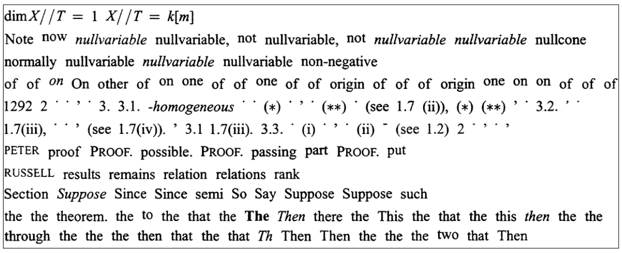
B. *Phase 2 - Displayed mathematical expression classification*

An image is represented in frequency domain by using Fourier transformation. It has been widely applied for image processing such as noise removal, pattern recognition, filtering and recently for *Image Quality Assessment* (IQA) [16]. For IQA, the transformation values of original and distortion images are compared to get the similarity scores.

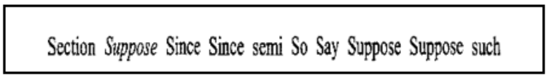
Given an image a with the size of $M \times N$ and its *Discrete Fourier Transform* (DFT) $A(\Omega, \psi)$, the mathematical equation of DFT [17] is

$$A(\Omega, \psi) = \sum_{m=1}^{M} \sum_{n=1}^{N} a(m,n) e^{-j(\Omega m + \psi n)} \tag{1}$$

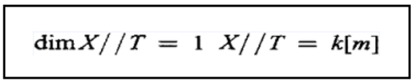
In practical, FFT algorithm calculates DFT more quickly, and the complexity of the algorithm is reduced from N^2 to $N \log_2 N$. After applying FFT algorithm to images, the phase and magnitude values are obtained. For instance, the FFT is applied for text image in Fig. 3b and displayed mathematical expression in Fig. 3c and phase and magnitude matrices are


dim$X//T$ = 1 $X//T$ = $k[m]$
Note now *nullvariable* nullvariable, not nullvariable, not *nullvariable nullvariable* nullcone
normally nullvariable *nullvariable* nullvariable non-negative
of of *on* On other of on one of of one of of origin of of of origin one on on of of of
1292 2 ˙ ˙ ’ ˙ 3. 3.1. -*homogeneous* ˙ ˙ (∗) ˙ ’ ˙ (∗∗) ˙ (see 1.7 (ii)), (∗) (∗∗) ’ ˙ 3.2. ’ ˙
1.7(iii), ˙ ˙ ’ (see 1.7(iv)). ’ 3.1 1.7(iii). 3.3. ˙ (i) ˙ ’ ˙ (ii) ¯ (see 1.2) 2 ˙ ˙ ’ ˙
PETER proof PROOF. possible. PROOF. passing part PROOF. put
RUSSELL results remains relation relations rank
Section *Suppose* Since Since semi So Say Suppose Suppose such
the the theorem. the to the that the **The** *Then* there the This the that the this *then* the the
through the the the then that the that *Th* Then Then the the the two that Then

(a) Input document image

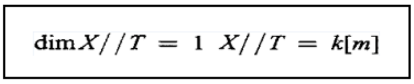


(b) Ordinary text line



(c) Displayed expression

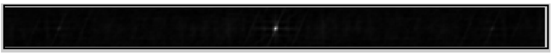
Fig. 3. An example of extraction from input document image



(a) Original image

(b) Reconstructed image by using Magnitude

(c) Reconstructed image by using Phase



(d) Reconstructed image by using Phase and Magnitude

Fig. 4. An example of reconstruction image

obtained. The average of FFT phase and magnitude values for text and expression images are (9.04;2.2650e-06) and (8.81;6.3568e-06), respectively.

Both phase and magnitude values are necessary to restore original image. However, it is known that phase value plays much more important role than magnitude [16]. It contains more perceptual information for human visibility than that of magnitude. This is shown in Fig. 4.

Intuitively, it is observed that the difference between ordinary text line and displayed expression based on the density of black pixels. Normally, the number of black pixels in ordinary text line is larger than in displayed expression. Therefore, *Power Spectrum* [17] is useful for comparing the density of images. The power corresponds with the FFT in Eq. (1) is shown as

$$E = \sum_{m=1}^{M}\sum_{n=1}^{N} |a[m,n]|^2 = \frac{1}{4\pi^2}\int_{-\pi}^{+\pi}\int_{-\pi}^{+\pi} |A(\Omega,\psi)|^2 d\Omega d\psi \quad (2)$$

For instance, Figure 5 illustrates the Power Spectrum of text line image in Fig. 3b and displayed expression image in Fig. 3c. It is shown that the density of text line is higher than that of displayed expression.

In order to have the overall quantitative measure for each image, FFT phase and magnitude values need to be combined. For simple computation, a linear function is used to combine the two values as

$$F_{score} = \alpha F_{phase} + \beta F_{magnitude}. \quad (3)$$

In this equation, F_{score} denotes the overall quantitative measure of an image. We determine the weight of phase value is greater than magnitude value due to their different importance as in above explanation.

In addition, MSE technique has been considered for a long time as an simple way to assess signal quality and fidelity. It is assumed that we have an original signal and its distorted one. MSE allows us to calculate difference between the two signals. For two images, MSE is often used to compare every pixel value of original image and distorted one. In this work, MSE is used for comparing the combination values of FFT phase and magnitude of two images. Given two images a_i and a_j with the same size of $M \times N$, $F(i)_{score}$ and $F(j)_{score}$ are calculated in Eq. (3), MSE value is as

$$MSE(i,j) = \frac{1}{M \times N}\sum_{m=1}^{M}\sum_{n=1}^{N}\left(F(i)_{score_{(m,n)}} - F(j)_{score_{(m,n)}}\right)^2 \quad (4)$$

Equation (4) measures the differences of displayed expression and text line images. The differences are normalized (called scores) and stored in two vectors, one is for displayed expressions and the other is for ordinary text. Basically, the two scoring vectors are formed as in the following.

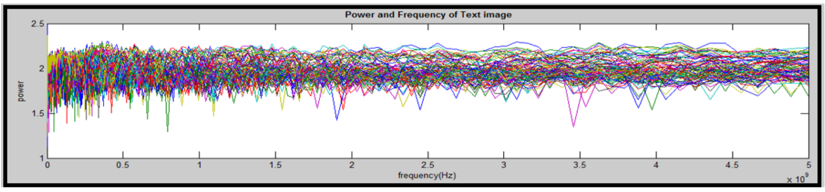
(1) Each item of scoring vector for displayed expressions is achieved by applying equation (4) for two images of displayed expressions.

(2) Each item of scoring vector for ordinary text is achieved by applying equation (4) for an image of ordinary text and that one of displayed expression.

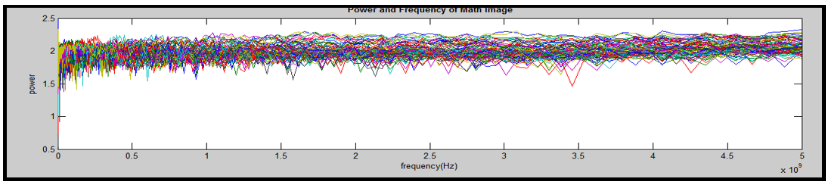
Then, these scoring vectors are used for the classification instead of feature extraction in previous methods. Both ad-hoc threshold and SVM are used as classifiers on these vectors. The threshold is set based on the average value of all elements of scoring vectors. In our work, images that have scores below the threshold are determined as displayed expressions and the others are text lines.

Besides, SVM is used for refined classification. This is a popular machine learning model to solve binary classification. This research employs SVM Light [11] that is an implementation of SVM in C language and is supported several interfaces for other programming languages such as Matlab, Python, Ruby. The diagram of the classification is shown in Fig. 6. In Fig. 6, training and testing steps are in blue and red, respectively. In the training step, scoring vectors computed by Eq. (4) with labels are fed into SVM. After that, trained SVM is used for classifying displayed expressions and text lines in the testing step.

978-1-5386-3210-9/17/$31.00 ©2017 IEEE



(a) Power Spectrum of an image of text line



(b) Power Spectrum of an image of displayed expression

Fig. 5. Example of Power Spectrum of an image of ordinary text line and displayed expression

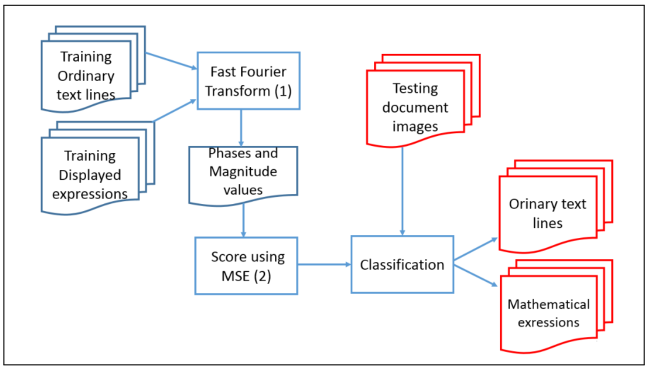
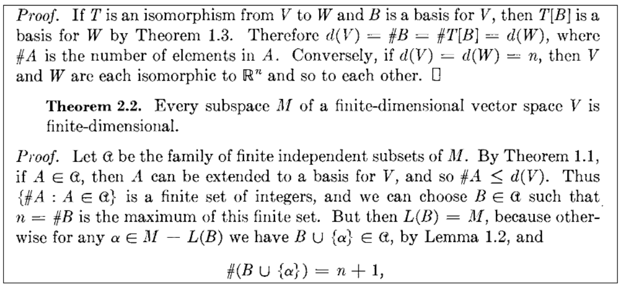


Fig. 6. Diagram of classification of ordinary text line and mathematical expression.



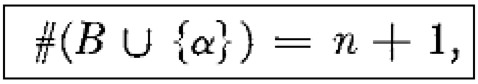
Proof. If T is an isomorphism from V to W and B is a basis for V, then $T[B]$ is a basis for W by Theorem 1.3. Therefore $d(V) = \#B = \#T[B] = d(W)$, where $\#A$ is the number of elements in A. Conversely, if $d(V) = d(W) = n$, then V and W are each isomorphic to $\mathbb{R}^n$ and so to each other. □

Theorem 2.2. Every subspace M of a finite-dimensional vector space V is finite-dimensional.

Proof. Let $\mathfrak{A}$ be the family of finite independent subsets of M. By Theorem 1.1, if $A \in \mathfrak{A}$, then A can be extended to a basis for V, and so $\#A \leq d(V)$. Thus $\{\#A : A \in \mathfrak{A}\}$ is a finite set of integers, and we can choose $B \in \mathfrak{A}$ such that $n = \#B$ is the maximum of this finite set. But then $L(B) = M$, because otherwise for any $\alpha \in M - L(B)$ we have $B \cup \{\alpha\} \in \mathfrak{A}$, by Lemma 1.2, and

$$\#(B \cup \{\alpha\}) = n + 1,$$

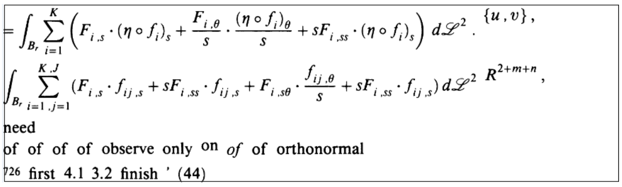
(a) An example of Harvard dataset document



$$\#(B \cup \{\alpha\}) = n + 1,$$

(b) An example of expression extracted correctly

Fig. 7. Mathematical expression extracted correctly from Havard dataset



$$= \int_{B_r} \sum_{i=1}^{K} \left(F_{i,s} \cdot (\eta \circ f_i)_s + \frac{F_{i,\theta}}{s} \cdot \frac{(\eta \circ f_i)_\theta}{s} + sF_{i,ss} \cdot (\eta \circ f_i)_s \right) d\mathscr{L}^2 \quad \{u, v\},$$

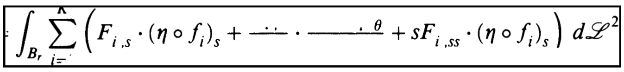
$$\int_{B_r} \sum_{i=1,j=1}^{K,J} (F_{i,s} \cdot f_{ij,s} + sF_{i,ss} \cdot f_{ij,s} + F_{i,s\theta} \cdot \frac{f_{ij,\theta}}{s} + sF_{i,ss} \cdot f_{ij,s})\, d\mathscr{L}^2 \quad R^{2+m+n},$$

heed

of of of observe only on of of orthonormal

726 first 4.1 3.2 finish ' (44)

(a) An example of Infty dataset document



$$\int_{B_r} \sum_{i=}^{K} \left(F_{i,s} \cdot (\eta \circ f_i)_s + \cdots \cdot \overline{\quad\quad ,\theta} + sF_{i,ss} \cdot (\eta \circ f_i)_s \right) d\mathscr{L}^2$$

(b) An example of expression extracted with errors

Fig. 8. Mathematical expression extracted with errors from Infty dataset

The overall steps of the proposed system are executed by the pseudo code in Algorithm 1.

V. Evaluation Result

A. Experimental results

For experimental purpose, 58 document pages of Havard and Infty datasets are randomly selected. 1254 ordinary text lines and 776 displayed expressions are extracted from these pages. For both Harvard and Infty datasets, we select 36 ordinary text line images and 36 displayed expression images for training; 642 images for testing.

Examples of text line extraction results are shown in Fig. 7 and Fig. 8. In Fig. 7, displayed expression is extracted correctly from a document page of Harvard dataset. In Fig. 8, a displayed expression is extracted with errors from a page of Infty dataset.

As the testing stage, text line extraction algorithm of OCRopus works stably for mathematical expressions those appear in only one line. However, mathematical expressions displayed in multiple lines are some times extracted with errors, it is necessary to correct manually for this classification purpose.

The confusion matrix of testing results on Infty and Harvard datasets are shown in Tables I and II, respectively. In our work, accuracy rate is calculated as

$$Accuracy = \frac{Correct\ Predictions}{Total\ Predictions} \tag{5}$$

Table III shows the performance comparison between the proposed method and other methods on the same datasets.

In general, classification results are lightly increased in comparison to the previous works [9] using Harvard dataset and [8] using Infty dataset. It shows the effectiveness of

TABLE I
CONFUSION MATRIX OF CLASSIFICATION METHOD BASED ON FFT AND SVM ON INFTY DATASET

Ground truth / Prediction	Expression	Text line	Total
Expression	184	10	194
Text line	28	420	448
Total	212	430	642

TABLE II
CONFUSION MATRIX OF CLASSIFICATION METHOD BASED ON FFT AND SVM ON HARVARD DATASET

Ground truth / Prediction	Expression	Text line	Total
Expression	209	6	215
Text line	3	424	427
Total	212	430	642

the proposed method. In addition, the implementation for [9] and [8] methods is carried out on these datasets for comparison, and the same SVM model is applied for testing as well. Better result comes from the combination of FFT and MSE. Moreover, hand-crafted features highly depend on properties of testing images and observer's knowledge. In contrast, images are objectively evaluated using the FFT and MSE. Thanks to this reason this method is stable across datasets.

In addition, in our testing, classification based on SVM is more accurate than that one based on threshold. Obtained classification accuracies are 94% and 92% for Infty dataset respectively. It is known that SVM classifies images into categories based on statistical learning theory. Training image feature is fed into SVM. Then, trained SVM separates testing images by optimizing the hyper-plane between two groups of images. This model is applied for many applications and shows robust strength. Meanwhile, threshold is set equal to the average of property values of images. It is able to quickly separate ordinary text and displayed images. However, in case that properties of ordinary text are similar to displayed image, these values are not much different, it may cause some confusions.

B. Discussion on particular errors

A particular misclassified case is shown in Fig. 9. In this case, an ordinary text line is misclassified to displayed expression. Errors of this case are caused by following main reasons:

(1) Text style is similar to mathematical style.

(2) Ordinary text lines contain major components of mathematical expressions.

TABLE III
ACCURACY COMPARISON OF THE PROPOSED METHOD

Method	Harvard Dataset (%)	Infty Dataset (%)
[9]	95	Not testing
[8]	Not testing	87.1
FFT+Threshold	Not testing	92
FFT+SVM	$97,6 \pm 1$	$93,9 \pm 1$

$V\ N\ T\ N\ 0\ X\ t\ X\ u\ v\ 0<\varepsilon,\gamma,\sigma,\tau<1.\ \langle T, NP, , F_i(x)\rangle\ (V, r)$

Fig. 9. An ordinary text line is misrecognized to displayed expression

VI. CONCLUSION AND FUTURE WORK

We have presented the method for extracting and classifying displayed mathematical expressions based on text line extraction, FFT, MSE and classification techniques. We have focused on FFT and MSE to get the difference between ordinary text and mathematical expression images in order to relieve the computation load for features as done in previous works. Better result is achieved as testing is carried out on the benchmark datasets in compared with previous works, and it shows the effectiveness of the proposed method.

In the future, inline mathematical expressions will be investigated. Accuracy can be increased by analyzing base line of images, then FFT is applied. In addition, some popular mathematical characters and functions are considered to be recognized to improve the accuracy.

REFERENCES

[1] "Tesseract," https://github.com/tesseract-ocr, accessed: 2017-07-14.

[2] "Ocropus," https://github.com/tmbdev/ocropy, accessed: 2017-07-14.

[3] "Infty," http://www.inftyreader.org/, accessed: 2017-07-14.

[4] "Abbyy," https://www.abbyy.com/, accessed: 2017-07-14.

[5] U. Garain, "Identification of mathematical expressions in document images," in *Document Analysis and Recognition, 2009. ICDAR '09. 10th International Conference on*. IEEE, 2009, pp. 1340–1344.

[6] C. Patel and A. Patel, "Optical character recognition by open source ocr tool tesseract: A case study," *International Journal of Computer Applications*, vol. 55, no. 10, 2012.

[7] F. Simistira, V. P. Marcus Liwicki, and B. Gatos, "Recognition of historical greek polytonic scripts using lstm networks," in *Document Analysis and Recognition (ICDAR), 2015 13th International Conference on*. IEEE, 2013, pp. 766–770.

[8] S. Yamazaki, F. Furukori, Q. Zhao, K. Shirai, and M. Okamoto, "Embedding a mathematical ocr module into ocropus," in *Document Analysis and Recognition (ICDAR), 2011 11th International Conference on*. IEEE, 2011.

[9] W.Chu and F.Liu, "Mathematical formula detection in heterogeneous document images," in *Proceedings of Conference on Technologies and Applications of Artificial Intelligence*. IEEE, 2013.

[10] A. C. Bovik and Z. Wang, "Mean squared error: Love it or leave it? a new look at signal fidelity measures," *IEEE Signal Processing Magazine*, vol. 26, no. 1, pp. 98 – 117, 2009.

[11] "SVM," http://svmlight.joachims.org/, accessed: 2017-07-14.

[12] "Havard dataset," http://www.math.harvard.edu/ shlomo/, accessed: 2017-07-14.

[13] M. S.Uchida, A.Nomura, "Quantitative analysis of mathematical documents," *International Journal on Document Analysis and Recognition*, vol. 7, no. 4, pp. 211– 218, 2005.

[14] H.Lee and J.Wang, "Design of a mathematical expression understanding system," *Pattern Recognition Letters*, vol. 18, no. 3, pp. 289 – 298, 1997.

[15] W. He, Y. Luo, F. Yin, H. Hu, J. Han, E. Ding, and C.-L. Liu, "Context-aware mathematical expression recognition: An end-to-end framework and a benchmark," in *Pattern Recognition (ICPR), 2016 23rd International Conference on*. IEEE, 2016, pp. 3246–3251.

[16] M. Narwaria, W. Lin, I. V. McLoughlin, S. Emmanuel, and L.-T. Chia, "Fourier transform-based scalable image quality measure," *IEEE Transactions on Image Processing*, vol. 21, no. 8, pp. 3364–3377, 2012.

[17] I. T. Young, J. J. Gerbrands, L. J. van Vliet, C. data Koninklijke Bibliotheek, D. Haag, Y. I. Theodore, G. J. Jacob, V. Vliet, and L. Jozef, "Fundamentals of image processing," 1995.