Mathematical Formula Recognition and Transformation to a Linear Format Suitable for Vocalization

Azadeh Nazemi
Electrical and Computer engineering Department
Curtin University
Perth, Australia
Azadeh.nazemi@postgrad.curtin.edu.au

Iain Murray
Electrical and Computer engineering Department
Curtin University
Perth, Australia
I.murray@curtin.edu.au

***Abstract*—Students with vision impairment encounter barriers in studying mathematics particularly in higher education levels. They must have an equal chance with sighted students in mathematics subjects. Making mathematics accessible to the vision impaired users is a complicated process. This accessibility can be static or dynamic, in static accessibility the user is presented with a representation of the entire mathematic expression passively such as using Braille, dynamic accessibility allows the user to navigate the mathematical content in accordance with its structure interactively such as audio format[1]. MATHSPEAK is an application that accepts objects described in LaTeX and converts it to a linear or sequential representation suitable for vocalization, describing functions to people with severe vision impairment. MATHSPEAK provides interactive dynamic access to mathematic expressions by rendering them to audio format. This paper describes a method to create plain text from images of mathematical formulae and convert this text to LaTeX which is used in the earlier developed algorithm, "MATHSPEAK".**

Keywords-component; LaTeX, Assistive Technology, Vision Impairment , Mathematic expression

I. INTRODUCTION

MATHSPEAK[2] is a software application which has been designed to represent math expression to vision impaired users via text to speech (TTS). MATHSPEAK aims to provide dynamic interactive representation of mathematic expressions. Screen readers and TTS read text in linear mode from left to right resulting in ambiguity and misrepresentation of the meaning contained in the formulae. A major problem in using TTS to read math's formulae is two dimensional natures of mathematical expressions which are not conveyed fully with screen readers. In addition, most of math's expressions are displayed as pictures, not text. Thus to be converted to speech they must be first re-rendered in a textual format.

Dealing with the large range of mathematical symbols in optical character recognition (OCR) systems is problematic. According to LATEX documentation there are about 500 symbols for mathematics expressions[3]. Many of the math's symbols are represented by non-alpha numeric symbols. Figure 1 shows some examples of these symbols.

%	+	=	<
>	*	∓	∞
≥	≤	⊄	⊅
⊆	⊇	⊈	⊉

Figure 1. Samples of non-alphanumeric math's symbols

The main OCR task is to capture information from non-text documents and save them for easy retrieval. Many of OCR systems capture only the text information and leave all or part of mathematical expressions unrecognized[4] and even if recognized still does not make sense with TTS. Figure 2 shows OCR outcomes for two sample images contain math's formula.

image	OCR outcome
$i-q'-\cdots-q^{(N-1)}$	could not find any text in this image
$X^{k}_{=0}$	X

Figure 2. OCR outcome for samples of math's formulae images

To solve the above issue regarding math's expression accessibility in non-text documents, a completed semantic recognition (Mathematic OCR or MOCR) and Math Expression Understanding System (MEUS) are required[8]. The integrated system must be able to convert math's formulae from non linear mode to linear mode. Table 1shows the samples for these two modes.

TABLE I. . COMPARISON, LINEAR AND NON LINEAR DESCRIPTIVE MODES

Non linear mode	Linear mode
$x = y^2$	x = y power 2
$z = \sum_{k=0}^{n} \mathrm{x}$	z= sum of x , from k=0 to n
$(1+x)^n = 1 + \frac{nx}{1!} + \frac{n(n-1)x^2}{2!} + \cdots$	$(1+x)^\wedge n = 1 + nx/1! + (n(n-$ $1)\,x^\wedge 2)/2!$
$\frac{-b \pm \sqrt{b^2-4ac}}{2a}$	$(-b \pm \sqrt{(b^\wedge 2-4ac)})/2a$
Y= $\int_a^b$ x	Y=integral of x ,over interval by a and b
Y=$\lim_{x\to\infty} x$	Y=limit of x, as x approaches infinity

II. MATHEMATICS RECOGNITION PROBLEMS

Ambiguity in mathematical expressions makes MOCR more complicated than normal OCR. Therefore strict ambiguity controlling is a critical issue in recognizing mathematics [5]. Ambiguity in mathematics area has various causes:

- In literary test, characters are sequentially read from left to right in the case of Latin languages. Math's expressions do not necessarily follow this rule. In some cases finding first element to be vocalized is not a trivial matter.
- Mathematic expressions are not linear. This issue is due to the problem of the relationship between symbols. Symbol semantic analysis is considered to solve ordering and relationship problems based on symbol role and situation concentration instead of symbol isolation.
- Some symbols for building formulae have different shapes in different situation but keep same meaning [6]. In these cases the established method of dictionary databases has been considered for recognizing mathematics symbols, has met with limited success. In math's formulae the meaning of the different elements depends on their shape and spatial occupied position. For example root symbol grows in height and width to fit its contents, or fraction bar grows in length to fit its numerator or denominator as shown in figure 3.These attributes create difficulties in the building of good classifiers for mathematical symbols. This problem has been addressed in both segmentation and symbol recognition modules in this research.

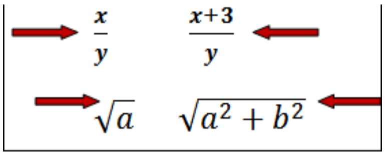


Figure 3. Math's symbols with different dimensions and common meaning

III. METHODOLOGY

The symbol context and meaning is ambiguous in mathematical expression as it is without grammatical or literary natural ordering. Based on this issue and complex nature of mathematics converting an image of a math's expression to text the steps shown in figure 4 must be performed:

- Preparation
- Primitive extraction
- English and Greek OCR
- MOCR
- Semantic analysis
- Merging primitives

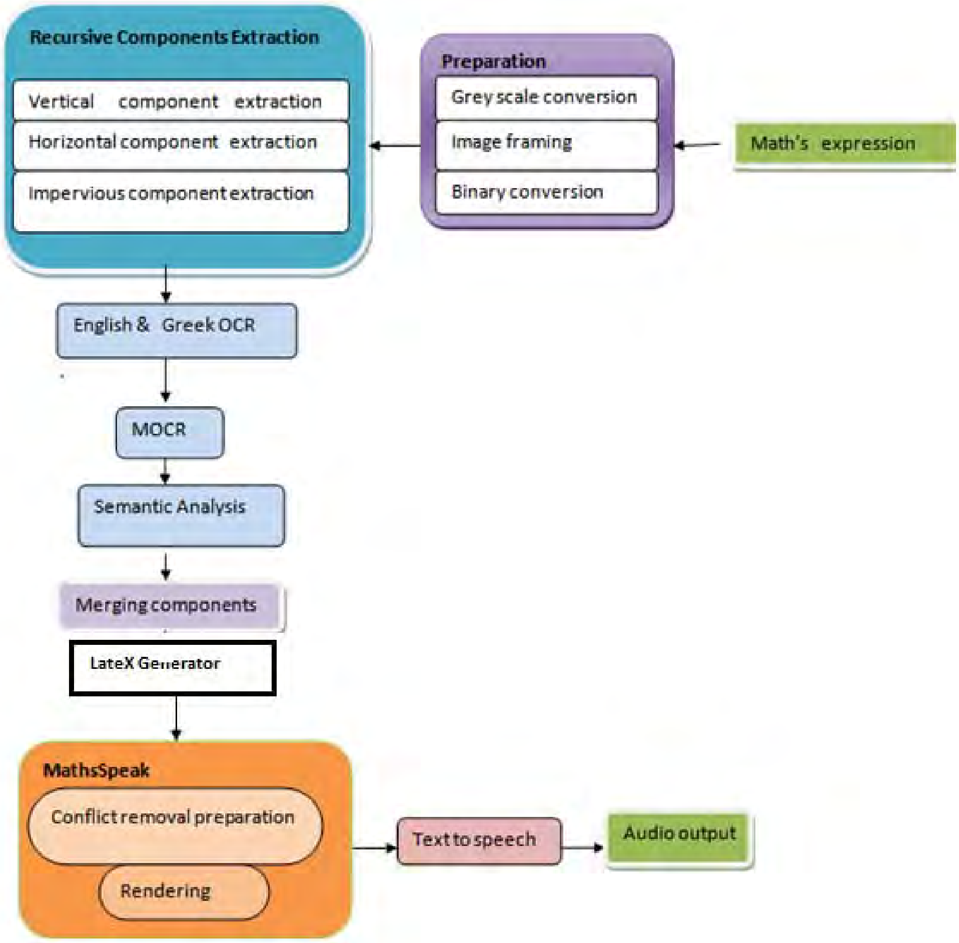


Figure 4. Flowchart of math's image conversion to text

A. *Preparation*

To reduce processing time, the preparation step is responsible for:

- Image scaling is considered to adjust the image size. Without scaling, large images have negative impacts to recognition performance [7]. Large images are not accurately converted to greyscale or binary images due to have large pixels population and necessary image processing for the large images is time consumable.

- Since an image that contains a formula may be in colour, greyscale or black and white, converting the expression image to greyscale has been considered. To convert colour image to greyscale, the color channels are mixed into a single grayscale image using a weighted combination [9]The weighted channels can be combined by simple addition combination. Images can have a variety of font sizes and may have a noise component. To remove noise, a trimming technique is used .This technique removes any edges that are exactly the same or nearly the same colour as the corner pixels.
- Remove white margins around image and frame it.
- Transform greyscale framed image to a binary image containing collections of black pixels. In performing two binary peaks (white and black pixel numbers), binary decision on majority population of pixels is made. It is assumed most of the pixels are black or white, and intermediate shades of grey are a minority. Therefore, the two peaks in a histogram should represent black and whites, and the midpoint between the two is an appropriate place to set the threshold. This assumption is normally correct for high quality images but for blurred or small images they may contains many pixels with different gray levels giving an indistinct outcome[7].
- Create good contrast between black and white

B. Primitives Extraction

Segmentation or Primitive Extraction is the separation of the components from the image by recognizing the spaces between them .The basic technique, termed Recursive Components Extraction (RCE), is used to solve two dimensional issues regarding math's expression [10]. RCE contains two sub modules: Vertical Components Extraction (VCE) and Horizontal Components Extraction (HCE).VCE first determines number of vertical segments in image by counting gaps (white pixels) and extracts these segments . HCE is applied for all VCE results.

To reduce ambiguity, at this point in the text array which consider to save final result , RCE adds open bracket "(" and close bracket ")" before and after the position of all obtained elements by HCE .If no HCE is possible, a single unit component is obtained, otherwise this procedure continues until all the single unit components are found. Once the image has been segmented and fully converted to single unit components , RCE generates a tree which the leaf nodes of the tree contain symbols associated with the image of math expression. Figure 5 shows an example for RCE procedure.

$\frac{x+\frac{x}{4}}{x-y}+x^2=\mathrm{z}$																							
$\frac{x+\frac{x}{4}}{x-y}$	+	x	2	=	z	VCE																	
$(x+\frac{x}{4})$	_____	(x-y)	HCE	+	x	2	-	-	HCE	z													
(	x	+	$\frac{x}{4}$	)	VCE	_____	(	x	-	y	)	VCE	+	x	2	-	-	z					
(	x	+	(	x	)	_	(	4	)	HCE	)	_____	(	x	-	y	)	+	x	2	-	-	z

Figure 5. RCE procedure for sample equation

RCE fails if symbols in an expression are both horizontally and vertically enclosed. Then neither VCE nor HCE can penetrate the expression to extract all components .To cover this issue before finalizing RCE Impervious Component Extraction (ICE) as a complementary function is applied to each leafs of tree. For the sample shown in figure 6 , ICE runs these steps to extract single unit components.

- Rotate the image 90 degree to the right
- Obtain maximum x
- Obtain minimum value for y when x is maximum value
- Split image horizontally from above obtained y
- Rotate two result images 90 degree to the left

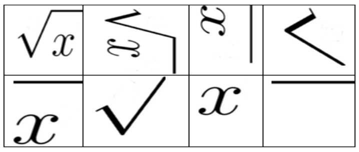


Figure 6. Impervious Component Extraction (ICE)

C. English and Greek OCR

All single unit components which are extracted by RCE are applied to English and Greek OCR in order from left to right one by one. Tesseract[11] an open source multi language OCR is able to convert image to text considering more than one language by using this command :

```
tesseract foo.png text  -l  eng + ell
```

Since many Greek alphabets are used in math's expression .Both English and Greek run at the same time for every single unit components. Checking Tesseract results shows several symbol are not recognized by the default OCR process. Therefore it is required to send them to the developed binary MOCR.

D. MOCR

All unrecognized segments by normal OCR include, special characters, symbols and some math operators (computational and relational) are converted to the image skeleton which contains the fewest pixels that keeps all the morphological properties of the image. This processing is necessary because the method used for MOCR is based on character morphology. Edges must be smoothed, rounded, and sharpened. Figure 7 shows generating skeleton of a symbol.

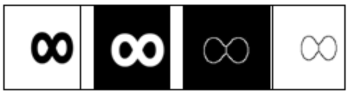


Figure 7. Symbol skeleton generation

All math symbols are stored in a dictionary specifically developed for this application. This database is completed by generating all non-alphanumeric math' symbol binary skeleton images with specific size(12) and font(DejaVu-Sanzs-Mono) using open source image processing package ImageMagick [12]. Each symbol in this database is represented by these fields:

- Description field contain symbol definition and determines symbol areas in mathematic scope
- Two digits binary number shows symbol symmetrical status
- Peripheral Direction Contributively (PDC) parameters[13] to recognize image statues such as complexion and connectivity. These parameters obtained by the counting the numbers of image crossing points by lines in the below equations (W=image width, H=image height)

x=W/2	x=W/3	x=W/4	y=H/2	y=H/3
y=H/4	y=W-x	y=H-x	y=W-H+x	y=H-W+x

All these properties must be calculated for each single unit component extracted by RCE which is not recognized by normal English and Greek OCR .These values are searched in database and be replaced with description field of most appropriate one. Finally the result of the component recognition process is, the best match, without considering the role of the symbol in expression. Table 2 compares PDC parameters in the original image of symbol component and image of symbol created by ImageMagick. Highlighted digits show common parameters in symbol image and created image in database.

TABLE II. SYMBOL PDC PARAMETERS

Symbol	**PDC parameters of Original image**	**PDC parameters of image created by ImageMagick**
∞	**1222224403**	**1222663213**
⊂	**2132221010**	**2141111010**
≻	**2133221010**	**2111111010**
≼	**3239221011**	**3263222020**
*	**3175441310**	**3124440110**
∫	**3113553020**	**3112220000**
s	**3133443020**	**3132221020**
√	**1244330210**	**1236330010**
_	**X110000000**	**X110000000**

E. Semantic Analysis

Accurate semantic structural analysis module helps to transform symbols consist two or more connected components and solve conflict problems, based on non-linearity and non-ordering nature of math's expression. Each single unit component is an image of pixels and

These points reveal the role of segment in expression in terms of “subscript/ superscript” or” from/to “ concepts such as:

- If $(x_{\max(i)}, y_{\min(i)}) = (x_{\min(i+1)}, y_{\max(i+1)})$ then (i) is powered by i+1
- If $(x_{\max(i)}, y_{\max(i)}) = (x_{\min(i+1)}, y_{\min(i+1)})$ then (i+1) is index of i

Subscript and superscript in the expression is reflected by adding ”index” and “power” to the description of expression.

For symbols such as Σ,Π , and Λ to recognize connected components , ordering of segments in the level of tree obtained by RCE module must be considered.

- If component (i) and (i+1) obtained by VCE and component (i)= Σ, Π ,Λ then component (i+1) is the starting of the expression that describes each term in Σ,Π , Λ

$\sum\square \prod\square \wedge\square$

- If component (i)= Σ,Π , Λ obtained by VCE and components (i+1) and (i+2) obtained by HCE then component (i+2) is the ending index and component (i+1) is the starting index

$\sum_{\square}^{\square}\square ,\prod_{\square}^{\square}\square ,\wedge_{\square}^{\square}\square$

- If component (i)= Σ,Π , Λ and components (i) (i-1) and (i+1) obtained by HCE then component (i+1) is the ending index and component (i-1) is the starting index

$\sum\limits_{\square}^{\square}\square ,\prod\limits_{\square}^{\square}\square ,\bigwedge\limits_{\square}^{\square}\square$

Determining two adjacent of each components help to define the role of component in expression and do accurate interpretation. For example during HCE ‘=’ symbol is split to two ‘-‘symbols, noting neighborhood property leads to get correct result and interprets two consecutive ‘-‘ symbols to “equal” instead of ” minus“. Information of segments for role recognition such as segments dimension (width, height) are collected in segment information database .

Height=maximum y-minimum y Width=maximum x-minimum x

Some similar symbols such as minus and fraction can be distinguished by having dimensions information. There are several mathematics area and many of mathematics symbols are used in more than one of these areas which include:

- Basic math
- Geometry
- Algebra
- Probability and statistics
- Set theory Logic
- Calculus and analysis

For avoiding ambiguity due to multi using symbol, a set of rules has been considered in MOCR based on comprehensive MUES. Table 3 shows some of these rules .

TABLE III. SEMANTIC ANALYSIS RULES

Symbol	Extracted by	Followed by	Meaning description	Example
^	VCE		Power	a^2
^	HCE		Normalized vector by length 1	$\hat{a}$
•	HCE		Derivative of	$\dot{x}$
•	VCE	•• with Similar starting points	Continue until	•••
•	HCE	•	Divide or ratio in basic math Field extension in set theory logic	a: b
•	HCE	• ;- ;- all extracted by VCR	Congruence	:=
•	VCE	•• if $y_{max\ (i)} < y_{min\ (i+1)}$	because	∵
•	VCE	•• If $y_{min\ (i)} >_{:} y_{max\ (i+1)}$	There fore	∴
_	HCE		Congruence	≡
~	VCE		Similar	
~	HCE		Median	Ȧ

*	VCE		Basic :multiplication in	2*3
			Calculus and analysis :convolves ion	A*B
			Linear algebra : Hermitian matrix	A^*
	VCE	|	Parallel	A||B

F. Merging and Plain Text to LateX Conversion

All single unit components which are converted to text by default OCR(Tesseract) and MOCR in merging or reassembling module from left to right are joined to each other by focusing situation and size properties saved in segment information database, using description field and make a text plain text, then plain text translate to LateX format using LaTeX Math Symbols database(Carlise,1995).

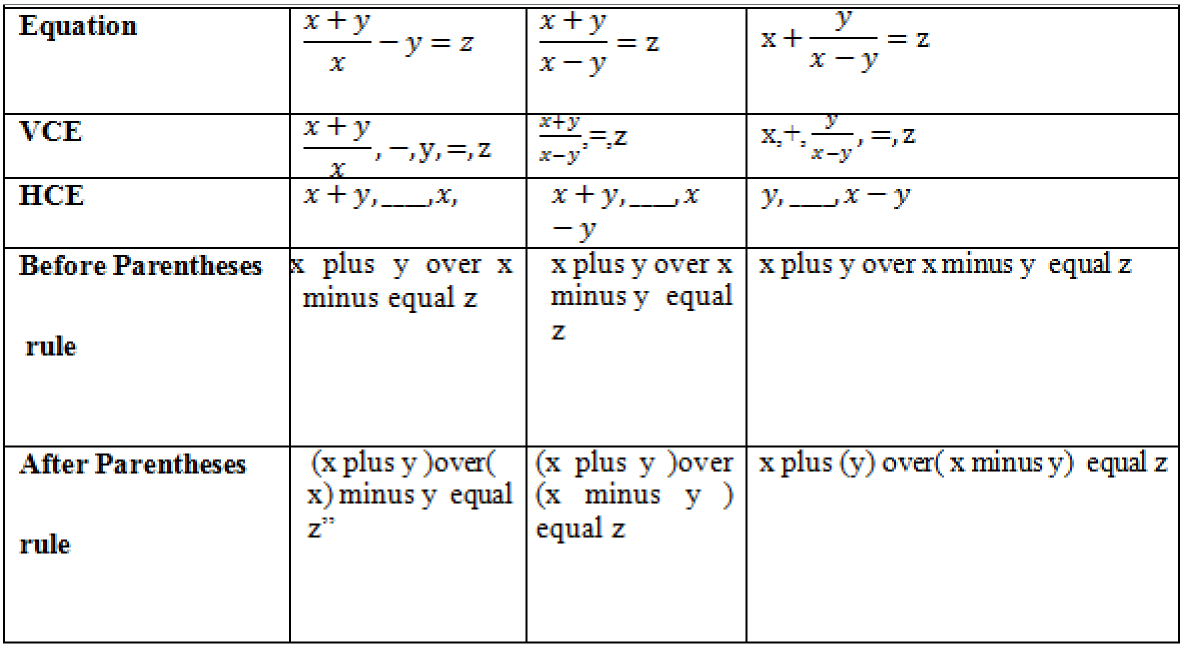
IV. Parentheses Rule

In this sample equation $\frac{x+y}{x-y}$ = z results of initial VCE are $\frac{x+y}{x-y}$ $, =, z$ then each of these components are sent to HCE results are x+y,___, x-y ,_ ,_ , z and the final output of RCE are x ,+,y,___,x,_,y,_,_,z MOCR matches + with plus ,___ with over and _,_ with equal. Therefore the linear result by concatenating components is ” x plus y over x minus y equal z”. This result cannot be appropriate and clear to send to next module, and it needs more consideration.

This equation x+$\frac{y}{x-y}$ $=$ z is totally different from first one .VCE results are: x, +, $\frac{y}{x-y}$, = ,z Results of HCE are x ,+,y ,___, $x - y$,_,_,z and final linear result is ” x plus y over x minus y equal z”

RCE by adding “(“ ,“)” in HCE step supports the system to keep and convey the original concepts of the equation. Table 5 shows this issue and the effect of applying parentheses rule to reduce ambiguity for a sample equation. In this method nesting in the expression is reflected by adding ”open bracket” and “close bracket” to the description of expression.

TABLE IV. . PARENTHESES RULE



Equation	$\frac{x+y}{x}-y=z$	$\frac{x+y}{x-y}=\mathrm{z}$	$\mathrm{x}+\frac{y}{x-y}=\mathrm{z}$
VCE	$\frac{x+y}{x},-,y,=,\mathrm{z}$	$\frac{x+y}{x-y},=,\mathrm{z}$	$\mathrm{x},+,\frac{y}{x-y},=,\mathrm{z}$
HCE	$x+y,\_\_\_,x,$	$x+y,\_\_\_,x-y$	$y,\_\_\_,x-y$
Before Parentheses rule	x plus y over x minus equal z	x plus y over x minus y equal z	x plus y over x minus y equal z
After Parentheses rule	(x plus y)over(x) minus y equal z”	(x plus y)over (x minus y) equal z	x plus (y) over(x minus y) equal z

V. CONCLUSION

Using RCE provides opportunity to represent simple and brief abstract of any math’s expression after running first VCE step , for example for this equation $\mathrm{y} = \sum_{k=0}^{n} \mathrm{x}$

The abstract will be: ”y is equal sum of x”.

Another feature of this method is dynamic interactive presentation for the mathematical content. Performing HCE supports navigation ability by generating hieratical nested sub expression structure of math’s expression. This research has been achieved to approach transformation image of math’s formulae to plain text to be convertible to LateX and applicable as input in MATHSPEAK. All considered rules in MOCR have been implemented and final generated results for about 580 math’s formulae in InftyMDB-1[14] database are clear ,comprehensive and contain the real concept. Future work is required to develop accurate math area detection in MUES .

REFERENCES

[1] A,Karshmer. G,Guptab. E,Pontellic.(2007). “Mathematics and Accessibility: a Survey” University of San Francisco , USA,University of Texas

[2] A, Nazemi .I, Murray. (2012) .”Mathspeak: An Audio Method for Presenting Mathematical Formulae to Blind Students”. Available at: http://hsi2012.debii.edu.au/

[3] D. Carlisle,(1995), LaTeX Math Symbols**,** Manchester University, Retrieved Feb 1, 2013, from : http://amath.colorado.edu/documentation/LaTeX/Symbols.pdf

[4] R, Fateman.T, Tokuyasu. B, Berman. N,Mitchell.(1995)” Optical Character Recognition for Typeset Mathematics”. Computer Science Division, EECS Department University of California at Berkeley. Journal of Visual Communication and Image Representation

[5] G,Erik . Miller .A. Paul .(1998).”Ambiguity and Constraint in Mathematical Expression recognition”. Viola Massachusetts Institute of TechnologyArtificial Intelligence Laboratory545 Technology Square, Office 707Cambridge, MA 02139 . In Proceedings of the 15'th National Conference on Artificial Intelligence (AAAI-98)

[6] P,Garcia. B, Co¨uasnon.(2002).”Using a Generic Document Recognition Method for Mathematical Formulae Recognition”. IRISA / INSA-D´epartement Informatique20, Avenue des buttes de Co¨esmes, CS 14315F-35043 Rennes Cedex, France, Graphics Recognition Algorithms and Applications Lecture Notes in Computer Science Volume 2390, 2002, pp 236-244

[7] L.R ,Gutiérrez.(2008)” Reconocimiento de texto matemático impreso”.Spain,Sevilla Publishing

[8] R,Haralick .T, Phillips. (1995).” Understanding Mathematical Expressions from Document Images” Dept. of Electrical Engineering, Unversity of Washington. Document Analysis and Recognition, 1995., Proceedings of the Third International Conference

[9] F,Weinhaus .(2012) .Fred Imagemagick scripts. Retrieved March 10, 2013, from http://www.fmwconcepts.com/imagemagick/

[10] A,Raja. M,Rayner. A,Sexton,A. V.Sorge,(2006)” Towards a Parser for Mathematical Formula Recognition “School of Computer Science, University of Birmingham, UK .

[11] Tesseract(2006). Retrieved March 1, 2013, from https://code.google.com/p/tesseract-ocr/wiki/ReadMe

[12] ImageMagick.(2002). Retrieved Feb 15 ,2013, from http://www.imagemagick.org/script/index.php

[13] C,Ishikawa, N.Ashida,Y. Enomoto,Y .Takata,M. Kimesaw,T. Kazuki. (2009), ”Recognition of Multi-Fonts Character in Early-Modern Printed Books” Nara Women’s University, Japan ,Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications, PDPTA 2009, Las Vegas,Vol2

[14] M, Suzuki. S,.Uchida, A, Nomura. (2005)” A Ground-Truthed Mathematical Character and Symbol Image Database**”** Fac. of Math., Kyushu Univ., Fukuoka, Japan .Document Analysis and Recognition, 2005. Proceedings. Eighth International Conference (pp.675-679)vol2