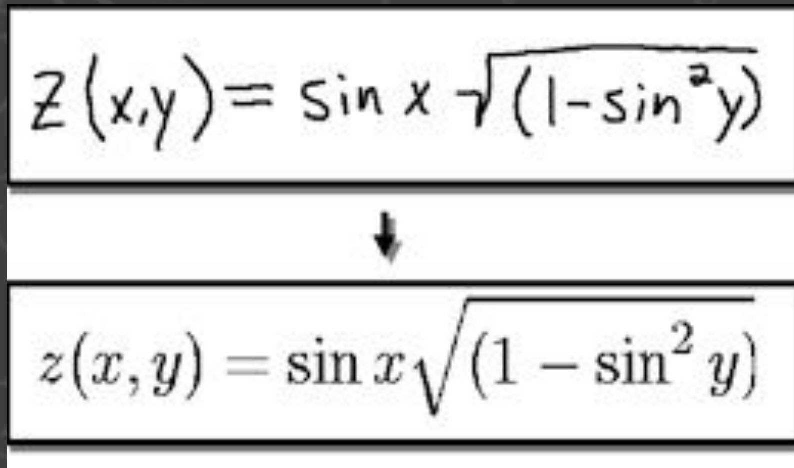
OFFLINE HANDWRITTEN MATHEMATICAL EXPRESSION RECOGNITION USING DEEP LEARNING TECHNIQUES

Ng Zheng Xun

Year 2 SCSE

U2020299H

16th November 2021



Handwriting

To

LaTeX

Learning LaTeX and manually conversion are time consuming and inefficient

Aid in digitalising of documents, learning for the visually impaired

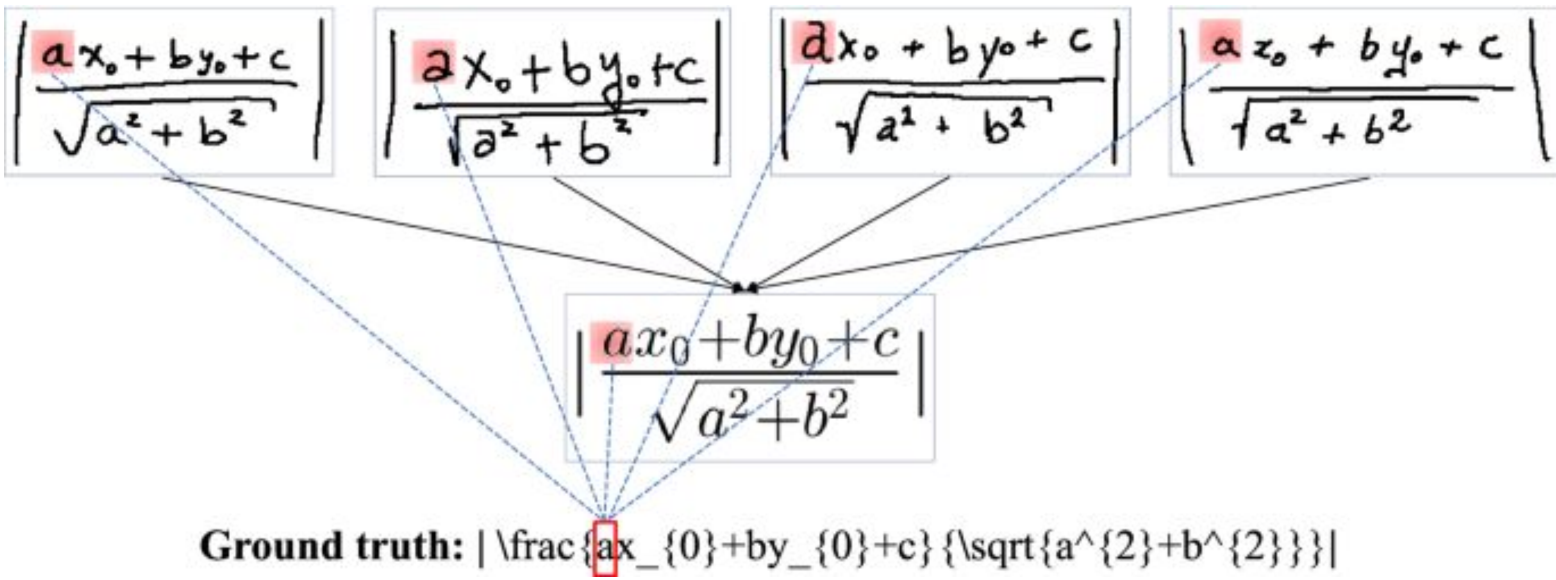
Handwritten Mathematical Expression Recognition (HMER)

Learning LaTeX and manually conversion are time consuming and inefficient

Aid in digitalising of documents,earning of the visually impaired

Challenges Faced in HMER

Variation of Handwriting Styles



Challenges Faced in HMER

Variation of Relative Scales

ground truth: 3 . 0 0 0 0 0 0 0 3

CROHME Dataset

Benchmark dataset for HMER

Data stored as InkML format

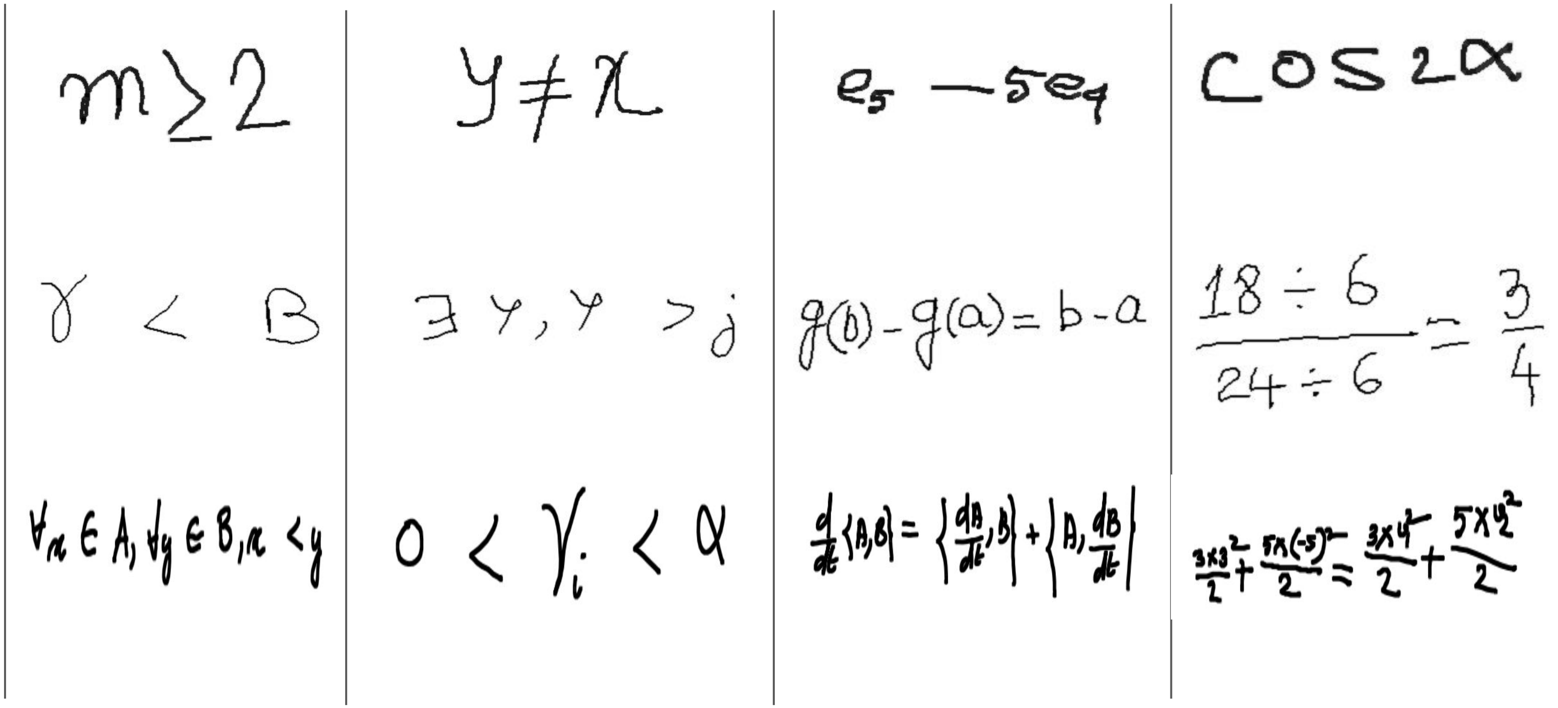
CROHME Dataset	Input Image Size	Dataset Size
2014	(256, 256, 3)	986
2016	(310, 310, 3)	10848
2019	(200, 515, 3)	11970

CROHME Dataset

CROHME 2014

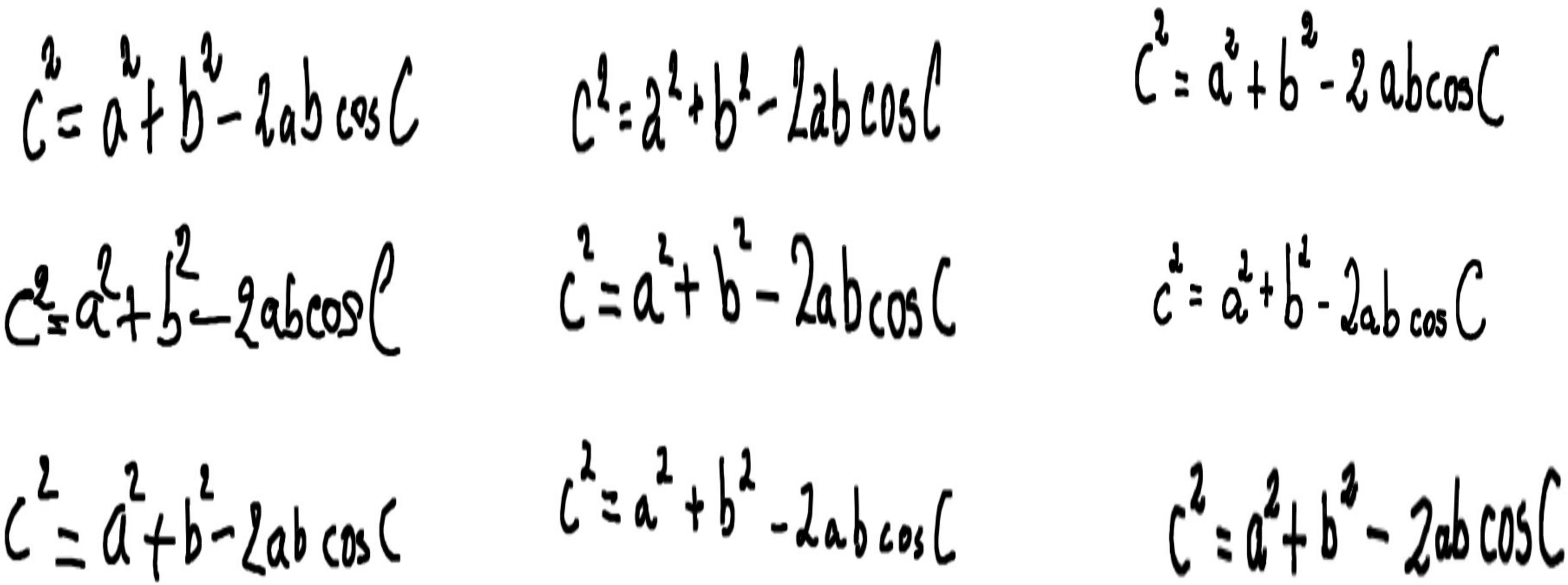
CROHME 2016

CROHME 2019



CROHME Dataset

$c^2 = a^2 + b^2 - 2ab \cos C$



Data Processing Pipeline

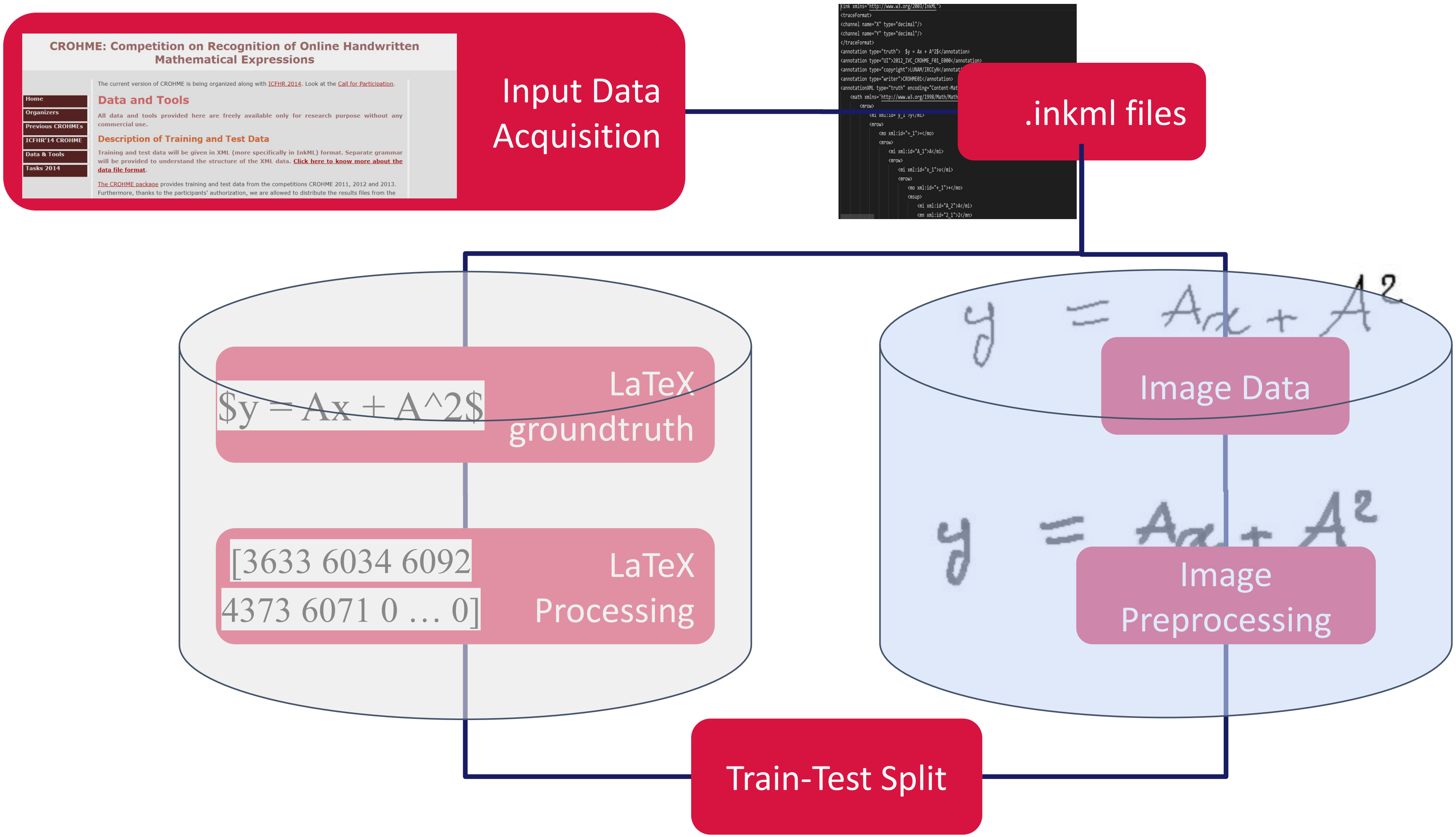
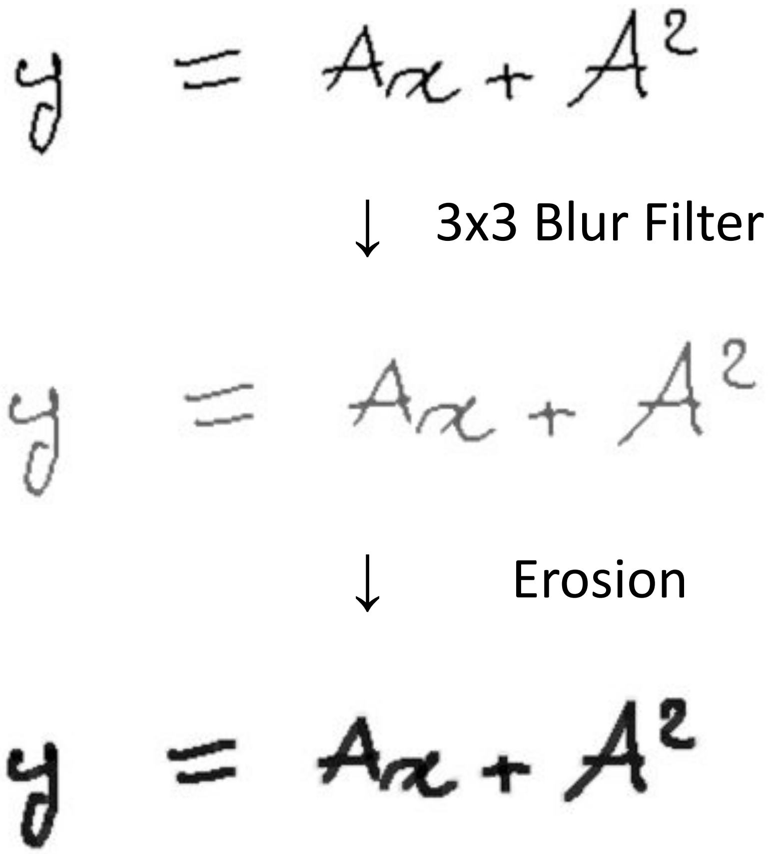

CROHME: Competition on Recognition of Online Handwritten Mathematical Expressions
Data and Tools
Description of Training and Test Data
Input Data Acquisition
.inkml files
$y = Ax + A^2$
LaTeX groundtruth
[3633 6034 6092 4373 6071 0 … 0]
LaTeX Processing
Image Data
Image Preprocessing
Train-Test Split

Image Preprocessing Pipeline



Various Methods to Encode Textual Groundtruth

$y = Ax + A^2$

‘$’ ‘y’ ‘=’ ‘A’ ‘x’ ‘+’ ‘A’ ‘^’ ‘2’ ‘$’

Character-level Encoding

‘$y' '=' 'Ax' '+' 'A^2$'

Subword-level Encoding

LaTeX Groundtruth Processing Pipeline

LaTeX Groundtruth Processing Pipeline

$y = Ax + A^2$

Groundtruth Segmentation

'$y' '=' 'Ax' '+' 'A^2$'

Tokenization

[3633 6034 6092 4373 6071]

Post-Padding

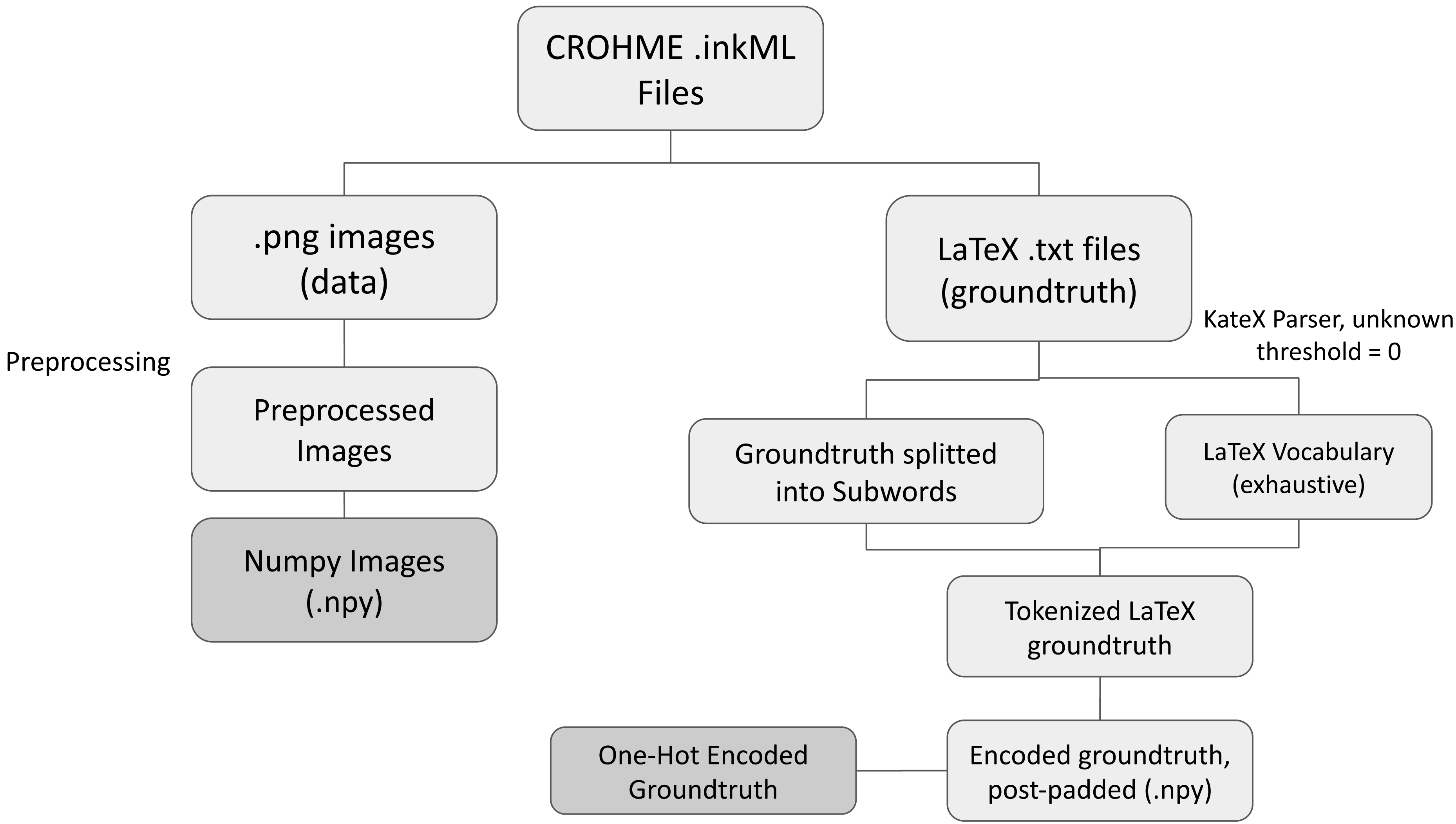
[3633 6034 6092 4373 6071 0 0 0 0….. 0]

One-Hot Encoding

[[[0. 0. 0. ... 0. 0. 0.], [1. 0. 0. ... 0. 0. 0.], [1. 0. 0. ... 0. 0. 0.]...
[1. 0. 0. ... 0. 0. 0.], [1. 0. 0. ... 0. 0. 0.], [1. 0. 0. ... 0. 0. 0.]

[[0. 0. 0. ... 0. 0. 0.], [0. 0. 0. ... 0. 0. 0.], [0. 0. 0. ... 0. 0. 0.]
... [1. 0. 0. ... 0. 0. 0.], [1. 0. 0. ... 0. 0. 0.], [1. 0. 0. ... 0. 0. 0.]]

[[0. 0. 0. ... 0. 0. 0.], [0. 0. 0. ... 0. 0. 0.], [0. 0. 0. ... 0. 0. 0.]
... [1. 0. 0. ... 0. 0. 0.], [1. 0. 0. ... 0. 0. 0.], [1. 0. 0. ... 0. 0. 0.]]]


CROHME .inkML Files
.png images (data)
LaTeX .txt files (groundtruth)
Preprocessing
Preprocessed Images
Numpy Images (.npy)
KateX Parser, unknown threshold = 0
Groundtruth splitted into Subwords
LaTeX Vocabulary (exhaustive)
Tokenized LaTeX groundtruth
One-Hot Encoded Groundtruth
Encoded groundtruth, post-padded (.npy)

Models and Methods

Various Libraries Used

TensorFlow

OpenCV

meierue/RNNLIB

A recurrent neural network library for sequence learning problems.

Keras

pytesseract

NumPy

pillow

scikit learn

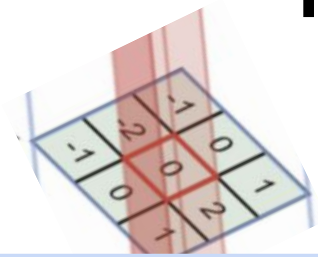
The Models

Model 1:
Keras Based Convolutional Neural Network (CNN)

Model 2:
Pre-trained Long-Short Term Memory (LSTM) - Tesseract 4

Model 3:
Pre-trained Recursive Neural Network SESHAT Parser

Keras CNN Model



Input Shape:
(310, 310, 3)

Convolutional

- 2D Convolutional Layer
- Max Pooling 2D Layer
- 2D Convolutional Layer
- Max Pooling 2D Layer
- 2D Convolutional Layer
- Batch Normalisation
- Max Pooling 2D Layer
- Flatten

Dense

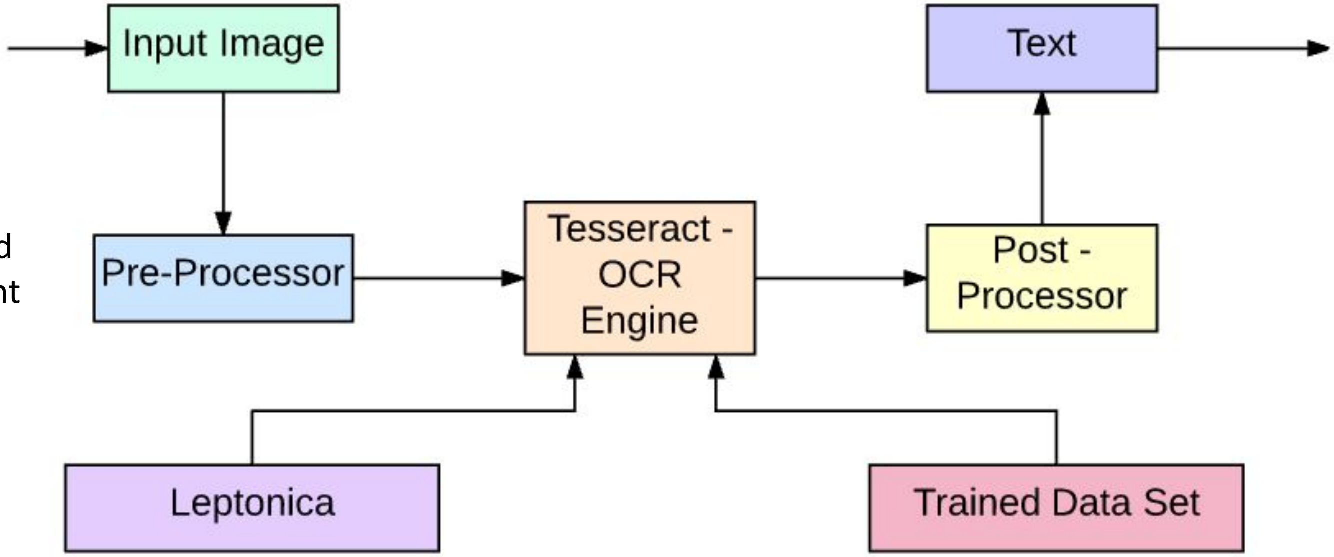
- Dense Layer - 128 neurons
- Dropout Layer - 0.5
- Batch Normalisation
- Softmax Layer - 53 neurons

[3633 6034 6092 4373 6071 0 0 0 0….. 0]

Tesseract 4 Model

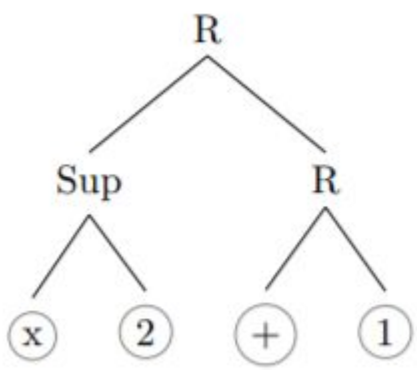
pytesseract

Connected
Component
Analysis

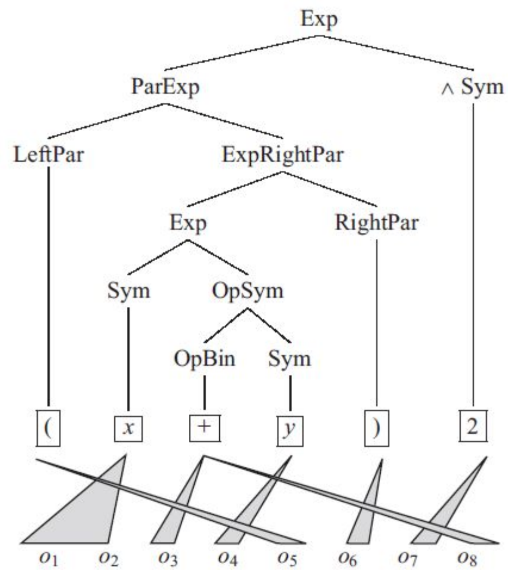


artificially generated corpus of images of text obtained from the web, in various fonts

SESHAT Parser Model



Relational tree for x^2 + 1



meierue/**RNNLIB**

A recurrent neural network library for sequence learning problems.

Sample SESHAT Output:

Number of strokes: 8

CYK table initialization:
Stroke 0:
rbracket [CloseBckt] 0.5698
3 [Digit] 0.0238538
\lambda [Greek] 1.17207e-05

Stroke 1:
dot [DecSep] 0.000441566
1 [Digit] 4.27196e-07

.....

Multi-stroke hypothesis: { 0 1 }
7 [Digit] 0.0760444
x [LetterMin] 0.107003

....

Most Likely Hypothesis (8 strokes)

Math Symbols: x { 0 1 }

LaTeX: x y COMMA Y \gt j

Results and Discussions

Character Error Rate (CER)

S = Number of Substitutions
D = Number of Deletions
I = Number of Insertions
N = Number of characters in groundtruth

$$CER = \frac{S + D + I}{N} \times 100$$

0 is the perfect score

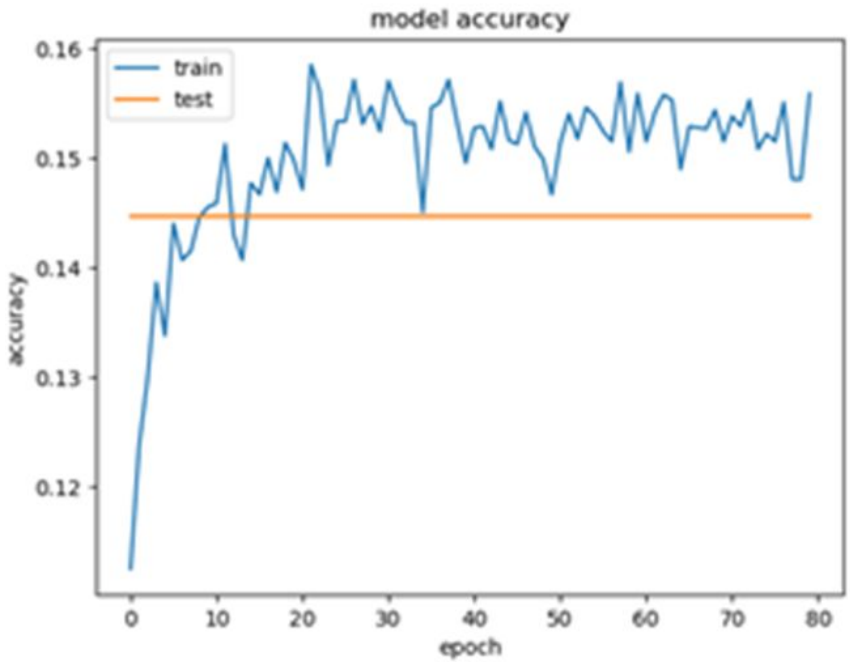
CER may exceed 100 if predicted expression is shorter than groundtruth

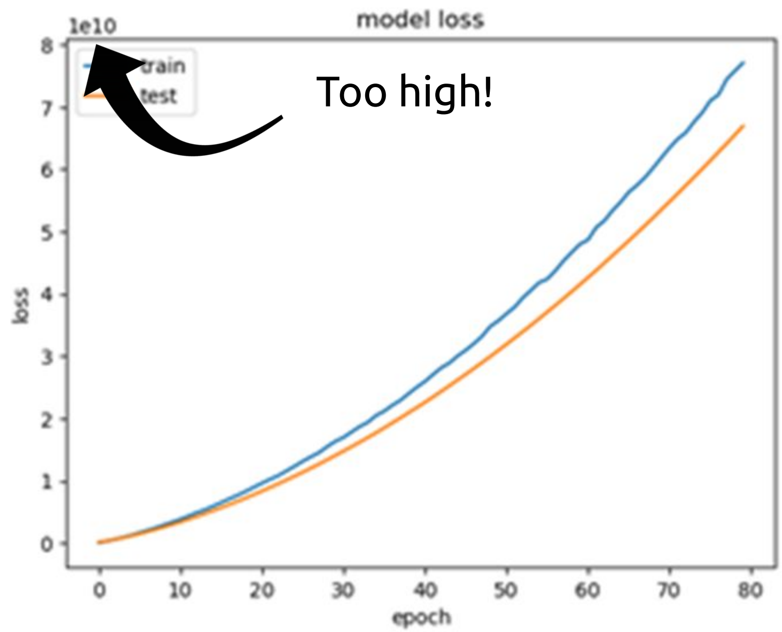
Results on CROHME datasets

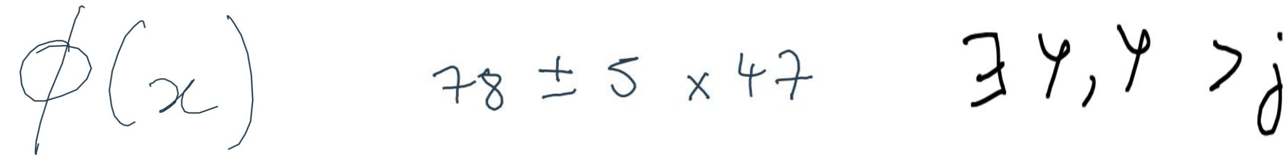
CROHME Dataset	Character Error Rate/CER (%)		
	Keras CNN	Tesseract 4	SESHAT Parser
2014	114.5	102.3	32.5
2016	111.9	102.8	35.7
2019	111.4	103.7	42.1
mean	112.6	102.9	32.7

A lower CER corresponds to a more accurate prediction

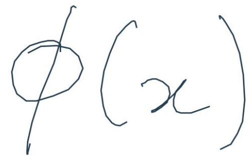
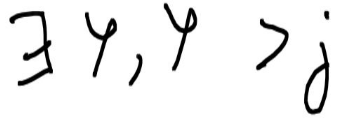
Keras CNN Plots



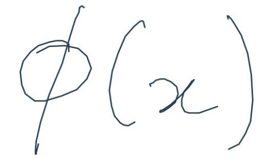




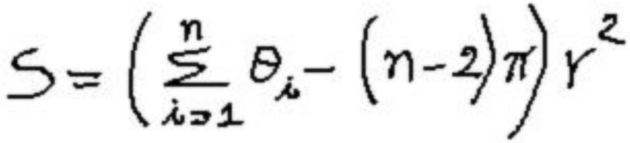
Groundtruth		∅(x)	78±5×47	∃ Y,Y > j
Groundtruth LaTeX		$\phi(x)$	$78 \pm 5\times 47$	$\exists Y, Y > j$
Keras CNN	Prediction (LaTeX)	$	$()	$(
	CER(%)	**400**	**300**	**850**

Groundtruth		∅(x)	78±5×47	∃ Y,Y > j
Groundtruth LaTeX		$\phi(x)$	$78 \pm 5\times 47$	$\exists Y, Y > j$
Keras CNN	Prediction (LaTeX)	$	$()	$(
	CER(%)	**400**	**300**	**850**
Tesseract 4	Prediction	P(x |	4g tS xb}	∃ ∀ |⋡⋡∂·
	CER(%)	**200**	**300**	**91.3**

			
Groundtruth		∅(x)	78±5×47	∃ Y,Y > j
Groundtruth LaTeX		\$\phi(x)\$	\$78 \pm 5\times 47\$	\$\exists Y, Y > j\$
Keras CNN	Prediction (LaTeX)	\$	\$()	\$(
	CER(%)	**400**	**300**	**850**
Tesseract 4	Prediction	P(x |	4g tS xb}	∃ ∀ |⊁⊁∂·
	CER(%)	**200**	**300**	**91.3**
SESHAT Parser	Prediction (LaTeX)	\$\phi (x)\$	\$78 \pm 5x 47\$	\$x y, Y \gt j\$
	Prediction	∅(x)	78±5x47	xy,Y j
	CER(%)	**0**	**25**	**71**

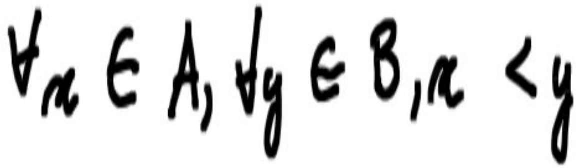
SESHAT's Mistakes

Generated Image		
Groundtruth	$A=B\timesC$	$S = \left(\sum_{i=1}^{n} \theta_i - (n-2)\pi \right) r^2$
SESHAT Parser Prediction	$A=Bxc$	$S = (\sum_{i=1}^{n} \theta_L - (n-2)\pi) r^2$

Understandable

SESHAT's Mistakes

Generated Image



Groundtruth

$\forall\ a \in A, \forall\ b \in B, a < b$

SESHAT Parser Prediction

$\forall\ ac - A, \forall\ bc - B, ab$

Some errors

Discussion on Computation Times

Keras CNN Model

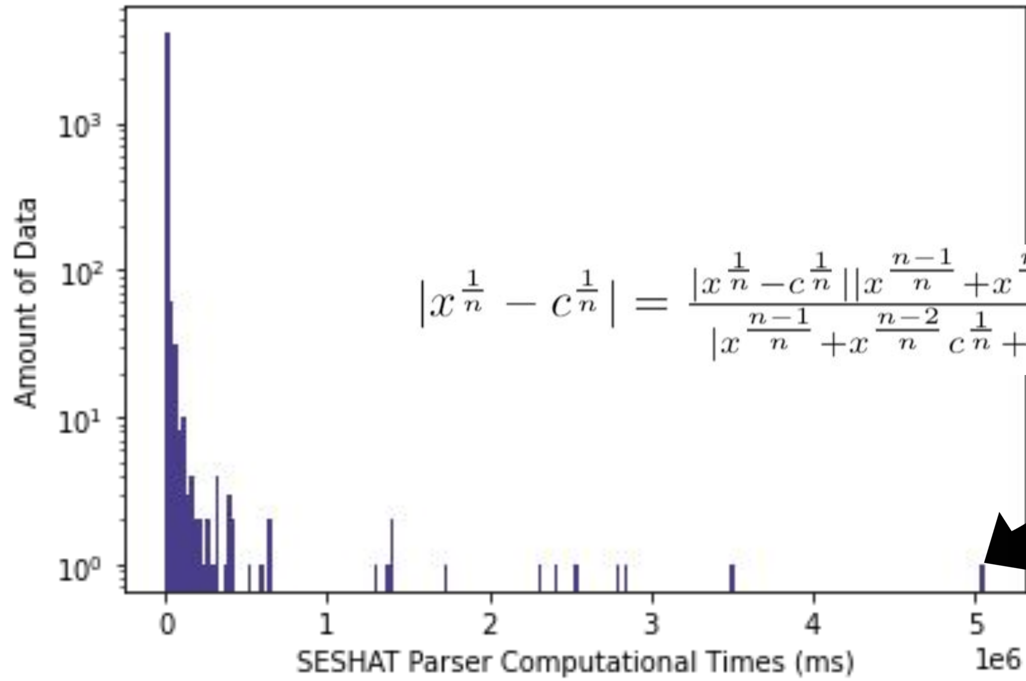
24 hours per 80 epochs
Evaluation within 5 mins

Tesseract 4 OCR Model

1 hour per 1k samples
Average 4 seconds per evaluation

SESHAT Parser

Discussion on Computation Times



$$\left|x^{\frac{1}{n}}-c^{\frac{1}{n}}\right| = \frac{\left|x^{\frac{1}{n}}-c^{\frac{1}{n}}\right|\left|x^{\frac{n-1}{n}}+x^{\frac{n-2}{n}}c^{\frac{1}{n}}+\cdots+x^{\frac{1}{n}}c^{\frac{n-2}{n}}\right|}{\left|x^{\frac{n-1}{n}}+x^{\frac{n-2}{n}}c^{\frac{1}{n}}+\cdots+x^{\frac{1}{n}}c^{\frac{n-2}{n}}+c^{\frac{n-1}{n}}\right|}$$

SESHAT Parser

In Conclusion

In Conclusion

Underlines how challenging and complex HMER is

SESHAT Parser is the much more Superior than other models trialed

Computational Times are not trivial, although SESHAT's timings (in ms) are promising for real-time applications

Further work could focus on more advanced methods such as image segmentation into sub expressions

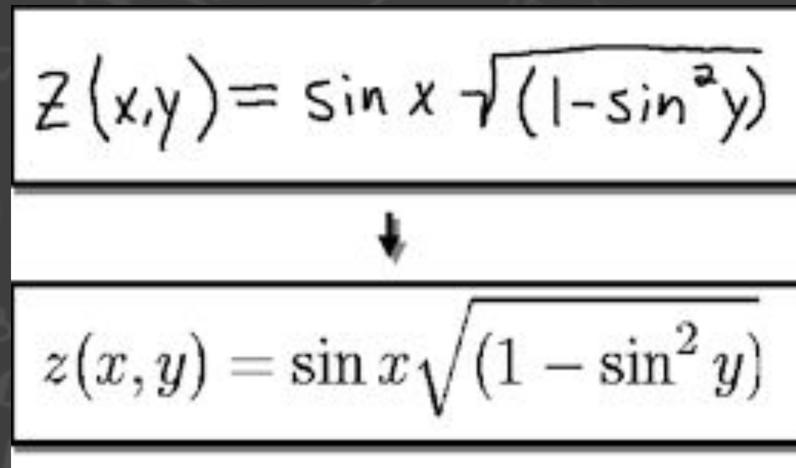
Acknowledgements

Research Supervisor and Mentor:
Dr Loke Yuan Ren,
School of Computer Science and Engineering, NTU

C.N. Yang Scholars Programme (CNYSP) Office

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Thank You!



Current manual methods are time consuming and inefficient

Several industrial and academic applications

Deep Learning approaches preferred

Keras CNN Model

hyperparameter	chosen values
Learning Rate	0.0001
Batch Size	32
Number of Classes	53
Number of Epochs for Training	80
Input Tensor Shape	(310, 310, 3)
Loss Function	Categorical Cross Entropy
Optimiser	Adam

Comparison with State of the Art

dataset/models	CROHME 2014	CROHME 2016	CROHME 2019
Offline SCAN + WAP	52.3	53.4	52.4
SESHAT Parser	90.5	91.7	100.0
Tesseract 4	102.3	102.8	103.7
Keras CNN	114.0	119.0	114.0