

**SISTEM PENGECAMAN TEKS TULISAN  
TANGAN LUAR TALIAN  
BERDASARKAN KAEDAH CNN**

Yew Rong De

Prof. Dr. Khairuddin Omar

*Fakulti Teknologi dan Sains Maklumat, Universiti Kebangsaan Malaysia*

**ABSTRACT**

Deep Learning (DL) is a hot topic in the introduction of current patterns and machine learning. DL solves many complex machine learning problems using different well-known libraries such as Tensorflow, Pytorch, Keras and Scikit-Learn. The availability of powerful pattern recognition tools creates great opportunities for future generations of smart applications. Today, there are still many industries that use handwriting text in management but lack of systems that can accurately identify text for many reasons, one of the main reasons being that handwriting text is unclear. Offline handwriting recognition (HTR) is intended to classify individual handwriting words so that handwritten text can be translated into digital. This model requires Python 3, Tensorflow, Numpy and OpenCV and Neural Networks (NN) with various architectures to train models that can correctly recognize the words. It consists of the Convolutional NN (CNN) layers, the Recurrent NN (RNN) layers and the last layers of Connectionist Temporal Classification (CTC). The data used in this study is data extracted from IAM. This data is an image of words written by different people to distinguish handwriting patterns. The total data used is 115321. The selected data will be divided into training data and test data. The accuracy of this trained HTR model is around 77%. The results also showed 89% accuracy in recognizing the words "he" and "her". Studies also show that CNN 7 layers in the neural network are better and more efficient than CNN 5 layers with a 1.4% character error rate increase.

## ABSTRAK

Pembelajaran Mendalam (DL) adalah topik hangat dalam pengenalan corak semasa dan pembelajaran mesin. DL dapat menyelesaikan banyak masalah pembelajaran mesin kompleks dengan menggunakan perpustakaan terkenal yang berbeza seperti Tensorflow, Pytorch, Keras dan Scikit-Learn. Ketersediaan alatan pengiktirafan corak yang kuat mencipta peluang yang besar untuk aplikasi pintar generasi akan datang. Pada masa kini, masih terdapat banyak industri yang menggunakan teks tulisan tangan dalam pengurusan tetapi kekurangan sistem yang dapat mengenali teks dengan tepat kerana banyak sebab, satu daripada sebab utamanya adalah teks tulisan tangan yang tidak jelas. Pengecaman teks tulisan tangan luar talian (HTR) bertujuan untuk mengklasifikasikan perkataan tulisan tangan individu supaya teks tulisan tangan dapat diterjemahkan ke dalam bentuk digital. Model ini memerlukan Python 3, Tensorflow, Numpy dan OpenCV dan Rangkaian Neural (NN) dengan pelbagai seni bina untuk melatih model yang dapat mengelaskan perkataan dengan tepat. Ia terdiri daripada aras Convolutional NN (CNN), aras Recurrent NN (RNN) dan aras terakhir Connectionist Temporal Classification (CTC). Data yang digunakan dalam kajian ini merupakan data yang diambil daripada IAM. Data ini merupakan imej perkataan yang ditulis oleh orang yang berbeza bagi membezakan corak tulisan tangan. Jumlah data yang digunakan adalah sebanyak 115321. Data yang dipilih akan dibahagikan kepada data latihan dan data ujian. Ketepatan model HTR yang dilatih ini mencapai sekitar 77%. Hasil juga menunjukkan ketepatan sebanyak 89% dalam mengecam perkataan "he" dan "her". Kajian juga menunjukkan CNN 7 aras dalam rangkaian neural adalah lebih baik dan lebih cekap berbanding CNN 5 aras dengan menunjukkan peningkatan character error rate sebanyak 1.4%.

## 1 PENGENALAN

Pengecaman teks tulisan tangan luar talian (HTR) merupakan kebolehan sebuah komputer atau alat peranti mengambil input tulisan tangan daripada sumber seperti bahan cetak, gambar dan alat-alat lain, atau menggunakan tulisan tangan secara langsung daripada skrin sentuh. Selepas itu, mentafsir tulisan tangan ini menjadi teks (Ofir Agasi, 2019).

HTR masih menjadi isu hangat bagi kebanyakan industri di Malaysia seperti bidang kesihatan, farmasi, insuran, perbankan dan juga perkhidmatan awam (Dimitri Schmitz, 2017).

Dalam HTR pada masa lepas, sistem mentafsirkan tulisan tangan pengguna ke format yang dapat difahami komputer seperti *Unicode*. HTR mempunyai banyak peringkat, bermula daripada pengecaman huruf yang mudah diteruskan dengan pengecaman seluruh kata dan akhir sekali pengecaman seluruh teks. *Apple's Newton* merupakan HTR yang pertama, tetapi ia dikira maju kerana ia telah mula mentafsir seluruh perkataan dan juga tulisan tangan yang kursif (Miika Silfverberg, 2007).

Walaupun teknologi pengecaman tulisan tangan tradisional telah berkembang, tetapi masih mempunyai kesusahan untuk mengecam tulisan yang kursif. Teknologi pengecaman tulisan tangan telah mempunyai penemuan baru hanya selepas kecerdasan mesin diperkenalkan.

Oleh itu, kajian ini akan menggunakan Rangkaian Neural(NN) dalam pembelajaran mendalam kerana pembelajaran mendalam adalah lebih bagus berbanding dengan kecerdasan mesin tradisional. Hal ini demikian kerana perbezaan prestasi di antara pembelajaran mendalam dengan kecerdasan mesin dari segi skala data yang semakin bertambah. Apabila skala kecil, pembelajaran mendalam tidak dapat memberi prestasi yang bagus kerana algoritma pembelajaran mendalam memerlukan skala data yang besar untuk memahami data dengan sempurna (Faizan Shaikh, 2017). Maka pembelajaran mendalam akan dipilih dalam kajian ini.

Set data yang digunakan dalam melatih model ini diambil daripada laman web IAM dan juga dikumpul sendiri. Sistem ini memerlukan kaedah *Python 3* dan *TensorFlow* untuk dijalankan (Harald Scheidl, 2018). Sistem ini akan menggantikan teknologi pengecaman tradisional.

## **2 PENYATAAN MASALAH**

Cabaran utama yang akan dihadapi dalam kajian ini adalah untuk mengecam tulisan tangan dengan tepat. Tulisan tangan tersebut akan melibatkan huruf, nombor dan juga simbol.

Masalah yang akan dihadapi ialah semua manusia mempunyai tulisan tangan yang berbeza. Tulisan tangan yang berbeza bermaksud huruf yang sama mempunyai cara tulis yang berbeza dan ini akan menyukarkan sistem untuk mengecam huruf tersebut.

Selain itu, kualiti gambar asal mungkin tidak jelas atau dokumen yang diisi tidak mempunyai tulisan tangan yang jelas atau tidak diisi dalam tempat yang disediakan menyebabkan kesusahan sistem untuk mengecam tulisan tersebut.

Terdapat kajian-kajian lepas yang telah mengeksplorasi pengecaman tulisan tangan

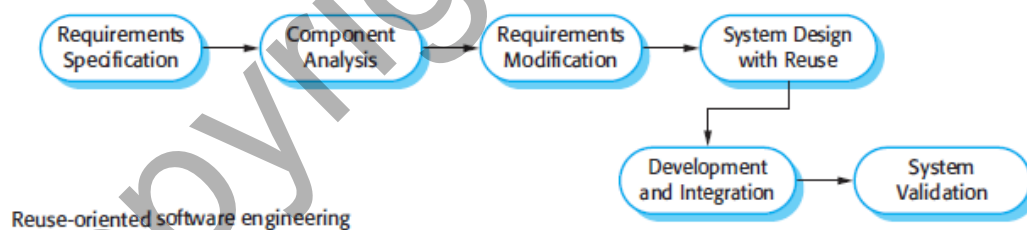
dengan menggunakan pelbagai cara, salah satunya adalah CNN, RNN dan CTC. Misalnya, terdapat banyak perisian atau sistem yang boleh mengesan teks tulisan tangan daripada gambar seperti *Optical Character Recognition (OCR)*. Walau bagaimanapun, menambah keupayaan HTR telah memberi cabaran yang besar bagi OCR (R. Reeve Ingle, 2019). Hal ini demikian kerana OCR hanya bekerja baik dengan gambar yang mempunyai kualiti tulisan yang bagus, ini bermakna dokumen yang bertulisan tangan tidak senang diproses dengan OCR (Shawn Radcliffe, 2019).

### 3 OBJEKTIF KAJIAN

Berdasarkan cadangan penyelesaian bagi kajian ini, objektif kajian yang akan dicapai ialah:

- I. Membangunkan sebuah sistem yang dapat mengesan tulisan tangan dan meningkatkan ketepatan sistem mengesan tulisan tangan dengan menggunakan kaedah CNN, RNN dan CTC.

### 4 METOD KAJIAN



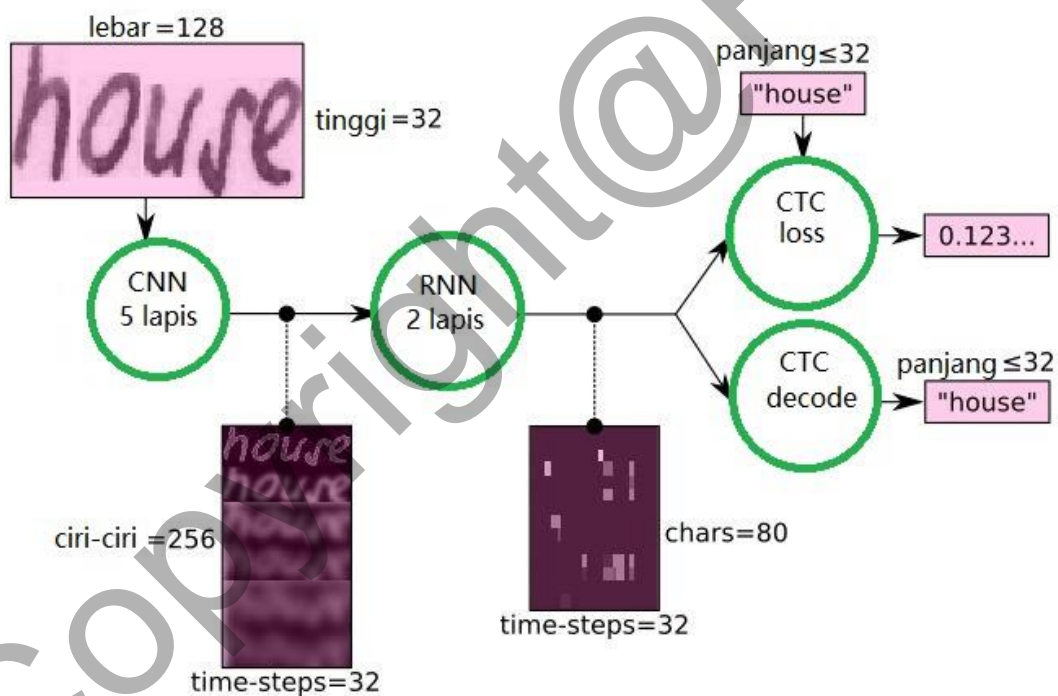
Rajah 1 : Fasa-fasa ROD

Berdasarkan Rajah 1, sebab kajian ini menggunakan ROD sebagai model proses ialah ROD dapat mengurangkan kos keseluruhan berbanding dengan model yang lain. Di samping itu, ROD dapat menjimatkan masa kerana setiap fasa dibina berdasarkan fasa lepas yang sudah ditentukan. ROD juga dapat meminimumkan ralat berlaku semasa dalam proses.

#### 4.1 Fasa Perancangan

Kaedah yang akan digunakan adalah model Rangkaian Neural(NN) dalam pembelajaran mendalam dimana ia akan dilatih dengan gambar tulisan daripada set data IAM. Sebab NN dipilih sebagai model kajian ini ialah semakin banyak data yang berbeza dan dilatih, maka semakin tinggi ketepatan sistem mengecam tulisan tangan. Pelaksanaan ini adalah perkara minimum yang diperlukan untuk menggunakan kaedah *TensorFlow* dalam HTR.

Dalam NN, setiap peringkat akan dijalankan dengan kaedah yang berbeza dan sebahagian kaedah akan diubah suai bagi memenuhi keperluan kajian. Hal ini demikian kerana kajian ini dimulakan dengan keperluan sumber yang minimum sahaja. Pengkaji akan melakukan pengubahsuaian bagi kod yang telah dibangunkan pada masa awal. Dengan metodologi ini, proses pembangunan kod akan dipercepatkan dengan bantuan kod masa awal dan pendapat daripada penyelia projek.



Rajah 2 : Gambaran Keseluruhan Sistem HTR

Berdasarkan Rajah 2, antara fasa-fasa yang dijalankan dalam kajian adalah berdasarkan kajian lepas yang bertajuk Membina Sistem Pengecaman Teks Tulisan Menggunakan Tensorflow (Harald Scheidl, 2018):

- I. CNN 5 aras: Setiap aras merangkumi 3 operasi. Pertama, *convolution operation*, kedua *non-linear RELU function* digunakan dan ketiga satu *pooling layer* akan meringkaskan rantau imej dan mengeluarkan satu versi input yang baru.
- II. RNN 2 aras: *Long Short-Term Memory(LSTM)* akan digunakan dalam aras ini kerana ia mampu menyebarkan maklumat dalam jarak yang panjang dan memberi lebih banyak sumber ciri-ciri latihan dalam model.
- III. Aras CTC: Aras ini akan mengira kehilangan nilai dan mengeluarkan teks yang telah dipastikan. Teks yang dikeluarkan tidak boleh melebihi 32 huruf.

## 4.2 Fasa Analisis

### 4.2.1 Spesifikasi Keperluan Sistem

#### - Keperluan Fungsian

Keperluan fungsian merupakan fungsi sistem perlulah berjalan dengan lancar. Fungsi merangkumi input, sikap sistem dan keputusan. Antaranya ialah:

- 1.) Model memproses imej dari set data untuk NN.
- 2.) Model membaca sampel, meletakkan mereka dalam kumpulan dan menyediakan antara muka iterator untuk memproses melalui data.
- 3.) Pengguna mengambil gambar teks tulisan tangan dan masukkan ke dalam sistem HTR.
- 4.) Sistem HTR memproses untuk mengecam teks tulisan tangan.

#### - Keperluan Perkakasan dan Perisian Pembangunan

Spesifikasi keperluan perkakasan diwujudkan untuk memastikan perkakasan yang digunakan mampu menampung perisian pembangunan sepanjang pembangunan sistem. Perkakasan yang dipilih perlulah mengikut perisian yang akan digunakan dalam fasa pembangunan di mana perisian yang berat memerlukan perkakasan berkuasa tinggi. Antara perkakasan dan perisian yang digunakan sepanjang pembangunan sistem ini ialah:

##### 4.2.1.1 Perkakasan Pembangunan

Dalam proses pembangunan sistem, komputer riba diperlukan untuk

membangunkan sistem HTR. Komputer riba dengan spesifikasi tertentu diperlukan supaya proses pembangunan dapat berjalan dengan lancar. Jadual 1 menunjukkan spesifikasi komputer riba yang akan digunakan sepanjang proses pembangunan. Selain itu, peranti mudah alih juga diperlukan dalam proses pembangunan sistem HTR. Jadual 2 menunjukkan spesifikasi peranti mudah alih yang digunakan sepanjang proses pembangunan.

Jadual 1: Spesifikasi Komputer Riba

<b><i>Model</i></b>	<i>Lenovo Legion</i>
<b><i>Operating System (OS)</i></b>	<i>Windows 10 Home Single Language</i>
<b><i>System Type</i></b>	<i>64-bit Operating System, x64-based processor</i>
<b><i>Processor</i></b>	<i>Intel® Core™ i5-8300H CPU @ 2.30GHz</i>
<b><i>RAM</i></b>	<i>8.00 GB (7.88 GB usable)</i>
<b><i>HDD Capacity</i></b>	<i>1TB HDD</i>
<b><i>SSD Capacity</i></b>	<i>256 GB</i>
<b><i>Graphic Card</i></b>	<i>NVIDIA GeForce GTX 1050 Ti</i>

Jadual 2: Spesifikasi Peranti Mudah Alih

<b><i>Model</i></b>	<i>Huawei nova 3i</i>
<b><i>Operating System</i></b>	<i>Android 9</i>
<b><i>Processor</i></b>	<i>Kirin 710</i>
<b><i>RAM</i></b>	<i>4.0 GB</i>
<b><i>Rear Camera</i></b>	<i>16-megapixel</i>

## 2) Perisian Pembangunan

### - *Google Colab*

*Google Colab* merupakan perkhidmatan awan percuma berdasarkan kepada *Jupyter Notebooks* yang menyokong *GPU* secara percuma yang dibangunkan oleh *Google*. *Google Colab* membenarkan sesiapa yang ingin membangunkan aplikasi atau sistem berdasarkan pembelajaran mendalam untuk menggunakan *PyTorch*, *Tensorflow*, *Keras* dan juga *OpenCV*.

## 3) Perkakasan Pengguna

Sistem HTR yang dibangunkan memerlukan teks gambar yang diambil peranti mudah alih. Oleh itu, pengguna perlu mempunyai peranti mudah alih dengan minimum spesifikasi supaya dapat memberikan keputusan yang tepat selepas diproses dalam sistem HTR. Jadual 3 menunjukkan spesifikasi minimum peranti mudah alih bagi pengguna.

Jadual 3: Spesifikasi Minimum Peranti Mudah Alih bagi Pengguna

<i>Operating System</i>	<i>Android / IOS</i>
<i>RAM</i>	<i>2GB</i>
<i>Rear Camera</i>	<i>8-megapixel</i>

## 4) Perisian Pengguna

### - Editor Imej

Pengguna memerlukan editor imej untuk edit teks gambar yang akan dimasukkan ke dalam untuk proses mengecam teks tulisan tangan. Hal ini disebabkan teks gambar perlu memenuhi kriteria tertentu seperti saiz supaya sistem HTR lebih senang untuk mengecam tulisan tersebut. Terdapat banyak perisian editor imej yang boleh digunakan oleh pengguna. Editor imej bagi peranti mudah alih adalah seperti *Adobe Lightroom*, *Aviary Photo Editor*, *Fotor Photo Editor* dan lain-lain (Joe Hindy, 2019). Manakala editor imej bagi komputer riba adalah seperti *Adobe Lightroom*, *Skylum Luminar*, *Adobe Photoshop* dan lain-lain (Nathan Singh, 2019).



## 4.3 Fasa Reka Bentuk

### 4.3.1 Reka Bentuk Pangkalan Data

Data yang digunakan dalam kajian ini merupakan data yang diambil daripada IAM. Data ini merupakan imej perkataan yang ditulis oleh orang yang berbeza bagi membezakan corak tulisan tangan. Jumlah data yang digunakan adalah sebanyak 115321. Data yang dipilih akan dibahagikan kepada data latihan dan data ujian.

Rajah 3, Rajah 4 dan Rajah 5 menunjukkan tiga contoh imej perkataan yang digunakan sebagai data.

A handwritten word 'life' in black ink on a white background. The letters are cursive and slightly slanted to the right.

Rajah 3: Contoh imej perkataan *life* - saiz 174 pixels x 63 pixels

A handwritten word 'from' in black ink on a white background. The letters are cursive and slightly slanted to the right.

Rajah 4: Contoh imej perkataan *from* - saiz 173 pixels x 72 pixels

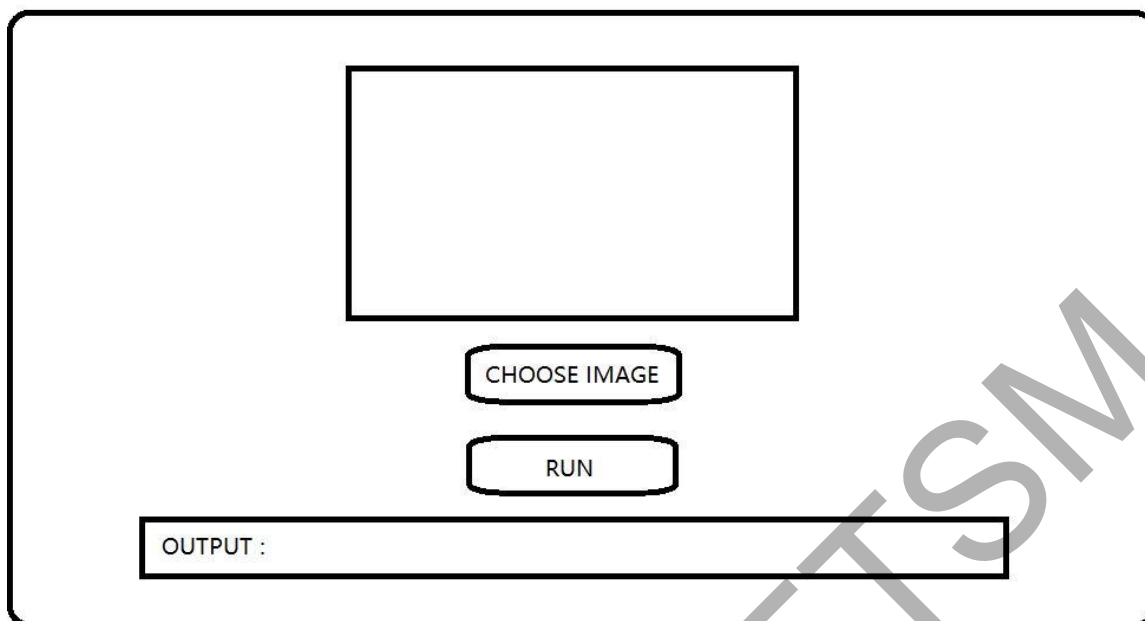
A handwritten word 'policy' in black ink on a white background. The letters are cursive and slightly slanted to the right.

Rajah 5: Contoh imej perkataan *policy* - saiz 250 pixels x 91 pixels

### 4.3.2 Reka Bentuk Antara Muka

Reka bentuk antara muka menjelaskan spesifikasi pengguna dalam bentuk lakaran. Dengan mereka bentuk antara muka, ia dapat memberi gambaran terhadap proses pembangunan sistem yang akan dibangunkan. Rajah-rajah di bawah akan menunjukkan antara muka sistem bagi kajian ini.

Rajah 6 menunjukkan antara muka bagi sistem yang akan dibangunkan.



Rajah 6: Antara muka sistem HTR

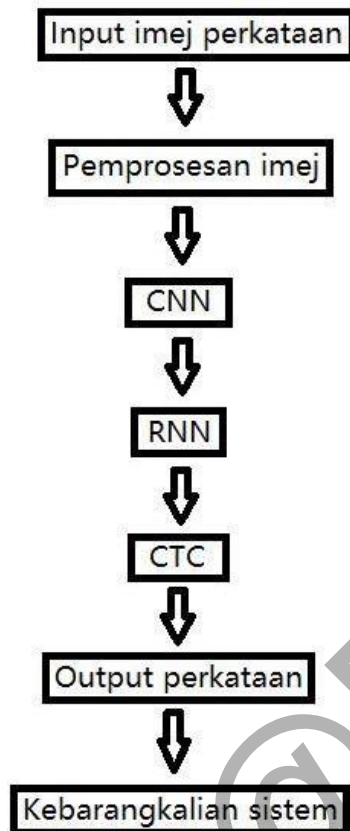
#### 4.3.3 Reka Bentuk Algoritma

Sistem kajian ini adalah untuk mengecam imej teks tulisan tangan bersama dengan Rajah 7 alur aliran algoritma yang diberi di bawah.

Selepas input imej perkataan, imej tersebut akan diproses kepada ketinggian 32 dan kelebaran 128 tanpa herotan. Imej yang telah diproses akan dimasukkan ke dalam 5 aras CNN. Setiap aras merangkumi 3 operasi. Pertama, *convolution operation*, kedua *non-linear RELU function* digunakan dan ketiga satu *pooling layer* akan meringkaskan rantau imej dan mengeluarkan satu versi input yang baru.

Kemudian, sistem akan bergerak kepada 2 aras RNN. *Long Short-Term Memory (LSTM)* akan digunakan dalam aras ini kerana ia mampu menyebarkan maklumat dalam jarak yang panjang dan memberi lebih banyak sumber ciri-ciri latihan dalam model.

Selepas itu, sistem akan bergerak kepada aras terakhir iaitu CTC. Aras ini akan mengira kehilangan nilai dan mengeluarkan teks yang telah dipastikan. Teks yang dikeluarkan tidak boleh melebihi 32 huruf.



Rajah 7: Alur aliran algoritma

#### 4.4 Fasa Pengujian

Kaedah yang digunakan untuk menguji model HTR adalah matrik kekeliruan. Rajah 8 menunjukkan report klasifikasi untuk model HTR. Dalam report klasifikasi tersebut, ketepatan sebanyak 89% dalam mengecam perkataan “he” dan “her”.

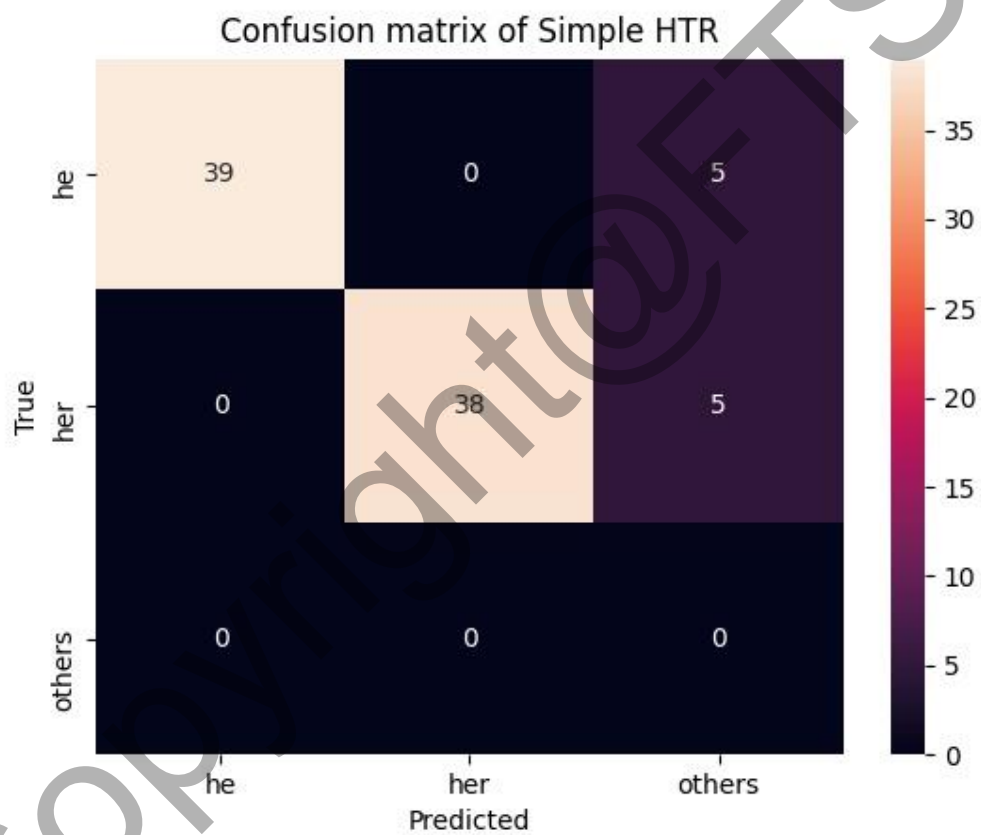
Berdasarkan pemerhatian, terdapat 2 sebab utama 11% perkataan “he” dan “her” tidak dapat dicam iaitu:

- 1.) Tulisan kursif
- 2.) Imej perkataan yang kabur

	precision	recall	f1-score	support
he	1.00	0.89	0.94	44
her	1.00	0.88	0.94	43
others	0.00	0.00	0.00	0
accuracy			0.89	87
macro avg	0.67	0.59	0.63	87
weighted avg	1.00	0.89	0.94	87

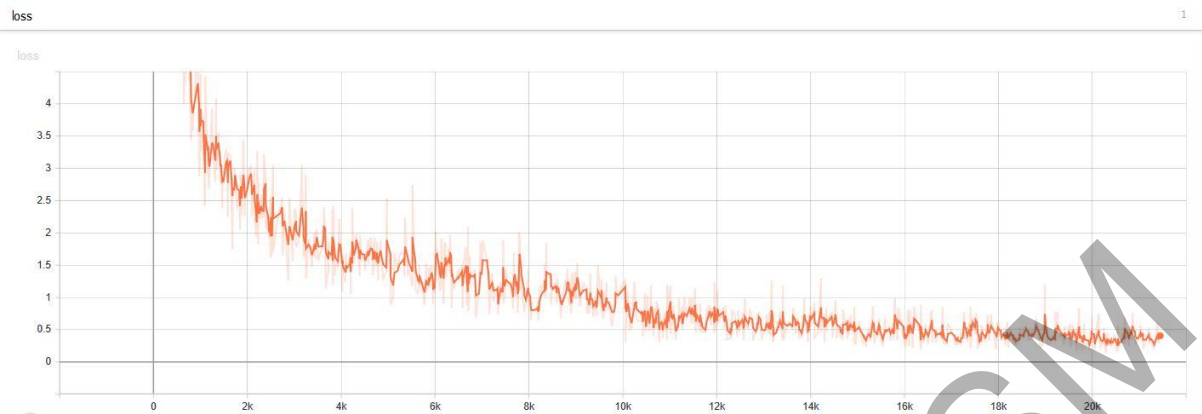
Rajah 8: Report klasifikasi

Rajah 9 menunjukkan matrik kekeliruan.



Rajah 9: Matrik kekeliruan

## 5 HASIL KAJIAN



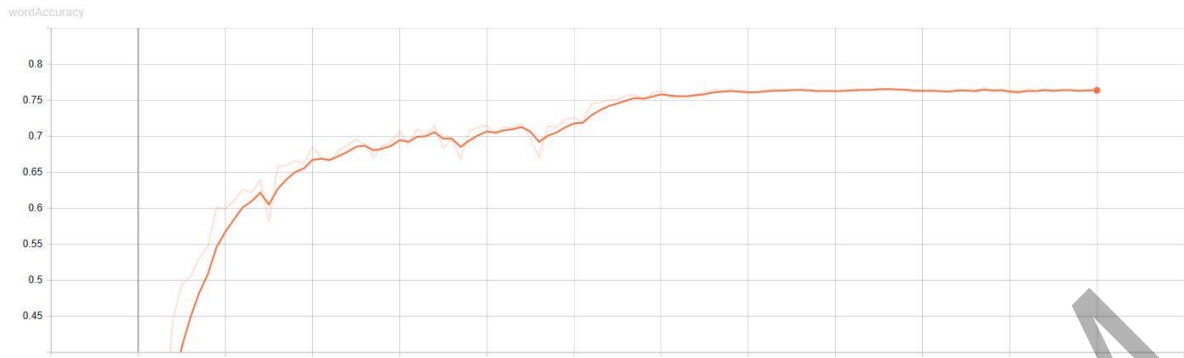
Rajah 10: *Loss* semasa latihan model

Rajah 10 menunjukkan *Loss* yang dikira semasa proses melatih model untuk sistem HTR. *Loss* merupakan kiraan kesalahan yang berlaku dalam setiap latihan dan pengesahan model dan tafsirannya adalah berdasarkan sejauh mana model yang dilatih dapat memberi keputusan yang bagus atau sebaliknya selepas setiap *epoch*. Maka, semakin kecil nilai *loss*, semakin tinggi ketepatan model sistem HTR.



Rajah 11: *Character Error Rate*

Rajah 11 menunjukkan *Character Error Rate* selepas model HTR siap dilatih. *Character Error Rate* menyatakan jumlah teks dalam tulisan tangan yang model HTR tidak dapat mengesan dengan betul. CER 10% bermaksud setiap aksara kesepuluh tidak dikenal pasti dengan betul. Aksara yang dinyatakan merangkumi huruf, tanda baca dan nombor.



Rajah 12: Ketepatan perkataan

Rajah 12 menunjukkan ketepatan model HTR yang dilatih. Berdasarkan rajah ini, model HTR mencapai ketepatan sekitar 77%.

Jadual 4 : Perbandingan Sistem HTR lama dengan baru

	Sistem HTR lama	Sistem HTR baru
<i>Character Error Rate</i>	11%	9.6%

Sistem HTR yang telah diubahsuai dan ditambahbaikkan telah membawa keputusan yang bagus berbanding kajian yang dirujuk. Maka, terbukti bahawa CNN 7 aras dalam rangkaian neural adalah lebih bagus dan lebih cekap berbanding CNN 5 aras. Jadual 4 menunjukkan character error rate telah meningkat sebanyak 1.4% dalam sistem HTR baru.

## 6 KESIMPULAN

Kesimpulannya, projek ini adalah pengubahsuaian sistem daripada kajian lepas yang bertajuk Membina Sistem Pengecaman Teks Tulisan Menggunakan Tensorflow (Harald Scheidl, 2018). Diharapkan sistem yang dibina ini akan dijadikan contoh bagi industri luar yang masih menggunakan tulisan tangan dalam pengurusan. Selain itu, diharap juga kajian ini akan dijadikan rujukan pada masa depan bagi memperbaiki sistem ini.

## 7 RUJUKAN

Ofir Agasi (2019). Handwriting Recognition (HWR).  
<https://www.techopedia.com/definition/196/handwriting-recognition-hwr>.

[9 September 2019]

Dimitri Schmitz (2017). The Challenge of Handwriting Recognition and Extraction.  
<https://www.moonoia.com/blog/handwriting-recognition>. [10 September 2019]

Miika Silfverberg (2007). English Language Handwriting Recognition Interfaces.  
<https://www.sciencedirect.com/topics/computer-science/handwriting-recognition>.

[11 September 2019]

FaizanShaikh (2017). Deep Learning vs. Machine Learning.  
<https://www.analyticsvidhya.com/blog/2017/04/comparison-between-deep-learning-machine-learning/>. [12 September 2019]

Harald Scheidl (2018). Build a Handwritten Text Recognition System using TensorFlow.  
<https://towardsdatascience.com/build-a-handwritten-text-recognition-system-using-tensorflow-2326a3487cd5>. [13 September 2019]

Nathan Singh (2019). 17 Best Photo Editing Software for Photographers.  
<https://enviragallery.com/best-photo-editing-software-for-photographers/>.

[11 Oktober 2019]

Joe Hindy (2019). 15 best Photo Editor Apps for Android for 2019.  
<https://www.androidauthority.com/photo-editor-apps-for-android-337494/>.

[11 Oktober 2019]