Comparison for Handwritten Character Recognition and Handwritten Text Recognition and Tesseract Tool on IJAZAh's Handwriting

Alexander Setiawan(✉), Kartika Gunadi, and Made Yoga Mahardika

Informatics Department, Faculty of Industrial Technology, Petra Christian University, Surabaya, Indonesia
{alexander,kgunadi}@petra.ac.id

Abstract. Handwriting is a form of being able to recognize various types of writing in various existing fonts. Unlike consistent computer letters, each human handwriting is unique in its form and consistency. These problems can be found in a document where the data is in the form of handwriting. Segmentation of the data location will use a run length smoothing algorithm with points as segmentation features. The Handwriting Text Recognition (HTR) technique requires segmented data into words. The Handwriting Character Recognition (HCR) technique requires segmented data into various characters. The process of this HCR technique uses the LeNet5 model using the EMNIST dataset. HTR uses the tesseract tool and a convolutional iterative neural network using the IAM database. Experiment on 10 samples of scan images, segmentation obtained an average accuracy of 95.6%. The HCR technique failed in the letter segmentation process in cursive handwriting. The easiest technique to use is the HTR with the helps of tesseract tool, tesseract tool also has a good performance. Tesseract managed to get word accuracy above 70% tested on 5 scan samples, 15 data fields.

Keywords: Handwritten Text Recognition (HTR) · Handwritten Character Recognition (HCR) · Segmentation · Tesseract

1 Introduction

Research on handwriting recognition, especially for Latin numbers and letters, is one of the topics in the development of pattern recognition techniques that are still developing today [1]. Writing recognition techniques can be divided into 2, namely character recognition and text recognition. The process of writing recognition from one way, by means of feature or feature extraction, and rocks. The two techniques have different approaches in carrying out writing recognition, especially in classification.

The problem that arises in carrying out the letter recognition process is how a recognition technique can regret various types of writing with different sizes, thicknesses,

*Please note that the AISC Editorial assumes that all authors have used the western naming convention, with given names preceding surnames. This determines the structure of the names in the running heads and the author index.

P. Vasant et al. (Eds.): ICO 2023, LNNS 729, pp. 1–10, 2023.
https://doi.org/10.1007/978-3-031-36246-0_95

and shapes [2]. This problem can be found especially in the case of handwriting. In contrast to computer letters that are consistent in their respective ways, the handwriting of every human being is unique. This problem can be found in the example of a diploma document which still uses handwriting in filling in the personal data of the owner.

These handwriting problems can be activated by applying writing recognition techniques. These techniques are character recognition and text recognition. Character recognition techniques using the Extended MNIST dataset. This dataset is a variant of the complete NIST dataset, called Extended MNIST (EMNIST), which follows the same conversion paradigm used to create the MNIST dataset [3]. Introduction to technical texts using the IAM dataset. The IAM database is a database of handwritten English sentences, which includes 1066 forms produced by approximately 400 different authors [4]. This scientific work aims to provide information in handwriting recognition, as well as an overview in the context of its application and performance.

2 Theoretical Basis

Run Length Smoothing Algorithm (RLSA) is a method used for block segmentation and text discrimination. RLSA is used for the blob process in binary images so that word or block segmentation can be done in the image. RLSA converts black (0) and white (1) pixels with the rule that all pixels in the original image are changed by 0 if the subsequent 1 pixel is less than or equal to a value.

The calculation of this algorithm is generally done 2 times, namely vertical then horizontal. This can be changed in the parameter if you only want to do any of the processing [5]. The rule iteration function already takes into account the calculation for horizontal, and can be used for vertical calculations as well. Transposing the image and using a horizontal iteration will get a vertical calculation. An example of the RLSA input and output is in Fig. 1.

Fig. 1. Run length smoothing algorithm output

2.1 Convolutional Neural Network (CNN)

In a neural network, a Convolutional Neural Network (CNN) is a model specially made for the categories of image recognition, image classification, object detection, face recognition, and others. CNN is a type of artificial neural network designed specifically for

processing pixel data [6]. CNN receives input in the form of images, is processed, and is classified into several categories (for example: cat, tiger, lion). CNN performs particularly well in looking for a pattern.

CNN has 3 main layers, namely the convolutional layer, the pooling layer, and the full connected layer. The convolution process in the convolution layer aims to extract features from the input image [7]. This process multiplies the image matrix with a certain size matrix filter and the product is summed to get the output feature. The convolution process is visualized as in Fig. 2.

Fig. 2. Convolution process

The pooling layer is a layer that reduces the dimensions of the feature map. The process at this layer is known as the step for down sampling. This is useful for speeding up computation because fewer parameters need to be updated and overfitting [8]. There are 2 types of pooling layers commonly used, namely max pooling and average pooling. Max pooling uses the maximum value per filter shift, while average pooling uses the average value. An example of the output pooling layer results can be seen in Fig. 3.

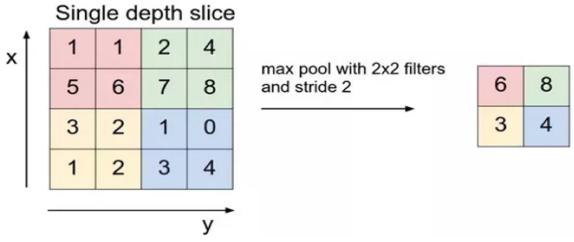

Fig. 3. Max polling

The fully-connected layer is a layer that is fully connected like an ordinary neural network. This layer will calculate the class score. Like a normal neural network, each neuron in this layer will be connected to all subsequent neurons in the volume. The flatten layer converts the matrix into a vector so that it can be fed to the fully connected layer. Example of flatten and fully connected layer results in Fig. 4.

2.2 Convolutional Recurrent Neural Network (CRNN)

Convolutional recurrent neural network (CRNN) is a neural network model that has a convolution layer, a recurrent layer, and finally a Connectionist Temporal Classification

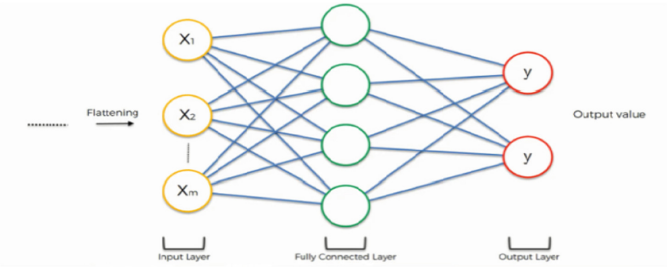

Fig. 4. Flattening & fully-connected layer

(CTC) layer. Unlike the usual CNN model, CRNN features extracted from the convolution layer are fed to the recurrent layer (not the fully connected layer) and the output layer uses CTC.

CTC is a layer function that is used to classify the final result into a string of a matrix. CTC is used as the final stage in a neural network sequence that aims to recognize characters. The results of the previous neural network are generally in the form of a matrix, this matrix is processed in order to obtain a text string as the final result of the existing series of processes. CTC is used as a transcription layer to translate the recurrent layer output matrix and basic truth text and calculate loss values. In conclusion, CTC feeds the initial raw output matrix of RNN and translates it into the final text. The length of the basic truth text and the recognized text is up to the maximum length of n characters from the training dataset. CRNN visualization can be seen in Fig. 5.

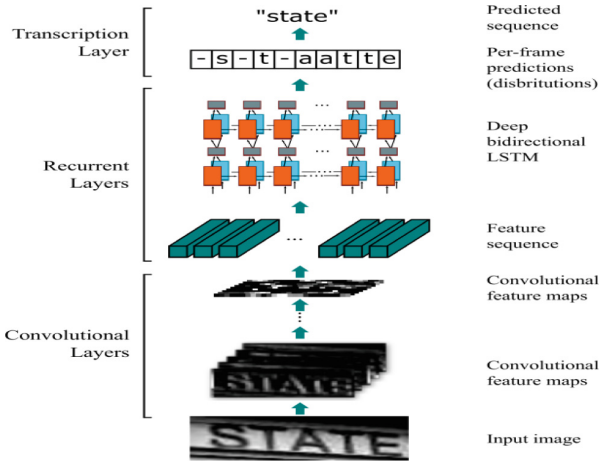

Fig. 5. Convolutional recurrent neural network

3 System Design

The system accepts input in the form of scanned images of elementary to high school diplomas in colour or grayscale. The system does not accept images from camera photos. Images from the camera still require a lot of preprocessing, to remove noise, perspective, and light and dark settings from documents. The recommended image input resolution is around or greater than or equal to 850 width $\times$ 1100 height, because the image will be pre-processed at that dimension. The output of the system is the result of OCR data in the form of a string.

This study will use the segmentation method with a point feature for segmentation of diplomas. This method is used to solve problems with diplomas that have different formats. The areas on the diploma are marked with dots, which are the places to fill in the data.

The image input will be carried out by a process called crop and reshape. This process is useful for uniform resolution of all diploma image input for the next segmentation process. This process affects the dot size and focus area of the input image. The image of the diploma will be cropped in the focus area and resized to a width of 650 and a height of 850. This process is subjective only for the diploma image. Flowchart process in Fig. 6.

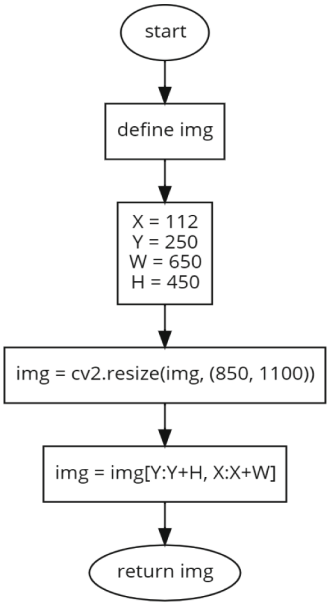

Fig. 6. Preprocess image flowchart

This process aims to obtain segmentation from the location of the data on the diploma. Dot segmentation or point segmentation uses the run length smoothing algorithm method and is used for the process of connecting vertically adjacent areas of the image. The dot

segmentation process actually only consists of 3 stages. Separating the dot location with the rest of the part and connecting it with RLSA method. Finally, segment the connected dot (line). Flowchart on Fig. 7.

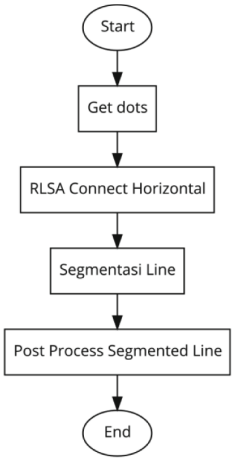

Fig. 7. Flowchart segment data location.

Before it can be fed into the model for recognition. Image data must first be processed. Each technique used requires a different process stage. Text recognition technique requires data to be segmented into images per word. Meanwhile, character recognition techniques require data to be segmented into letters/numbers.

The letter segmentation failed in cursive writing, where in this study an adjustment algorithm was made. The process of segmenting letters only uses regular contour searches in the process. The letter image is processed to resemble the image on the MNIST dataset measuring 28 $\times$ 28 and centerized. Process flowchart in Fig. 8.

4 Implementation System and Testing System

Testing must be done, and the purpose of the test is to receive feedback from participants so that the level of usefulness of this application can be determined [9]. The results of word segmentation will be processed to be predictable in the next process. The segmentation results will be processed again to be able to feed into the model for prediction. Using the same function as the image processing function for this text recognition model training. The image will be converted into grayscale and the aspect ratio changes according to the shape model input. The programming language used is Python. Machine learning API is used to detect images (Cloud Vision API) such as logos or photos, video, natural language (text), voice, and translators [10]. The input image must be grayscale with a size of 128 $\times$ 32 pixels, white on text and black on the background (invert). An example of changing the image size to 128 $\times$ 32 can be seen in Fig. 9.

Testing the accuracy of the RLSA segmentation results with point features compared to text features. Segmentation aims to obtain all pieces of data separately. The calculation

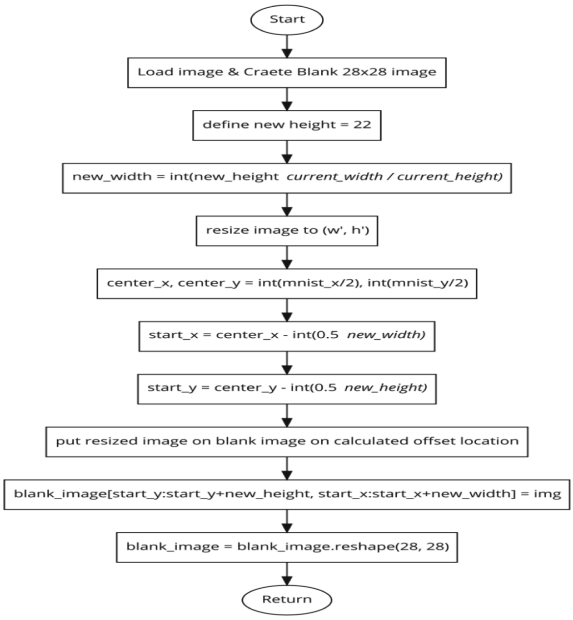

Fig. 8. Convert image to MNIST format flowchart

Fig. 9. Preprocessing in text recognition

of accuracy uses 10 sample diplomas that have been selected on the grounds that they have a fairly good standard of image quality and are also finding cases that can reduce the accuracy of segmentation.

From each sample, segmentation is carried out, and from each segment the segmented character will be calculated how many correctness of the segmented character is divided by the number of slots in the certificate sample form. The final result of each sample is added and divided by the number of existing samples. The test table can be seen in Table 1. Visualization of the final segmentation results can be seen in Fig. 10.

Testing was carried out on 5 diploma samples in 3 parts of the data. The diploma has been specially chosen because it has quite good image quality and has cases that affect segmentation and recognition performance. The test was carried out with the LeNet5 character recognition model trained with Extended MNIST data. The text recognition model uses 5 layers of convolution and 2 layers of bidirectional LSTM 256 units, which are trained with the IAM database. The tesseract tool is also used for text recognition.

Table 2 is a test of the accuracy of the model reading the name part of the sample. The name on the diploma is mostly in capital letters. This test aims to find out how well the model reads handwriting with capital letters. This makes the character recognition model (Extended MNIST) have quite high accuracy because letters can be segmented quite well.

Table 1. Segmentation accuracy on 10 diplomas.

Sample	RLSA-dot	RLSA-text-5
ijazah1.jpg	0.98	0.86
ijazah2.jpg	0.85	0.33
ijazah3.jpg	0.95	0.55
random1.jpg	0.97	0.50
random2.jpg	1.00	0.62
random3.jpg	1.00	0.90
random4.jpg	1.00	0.87
random5jpg	0.77	0.22
random6.jpg	1.00	0.45
Random7.jpg	0.96	0.57
ACC	0.951	0.587

The character recognition model also has higher accuracy than the text recognition model (IAM). Based on observations, the IAM model is trained with not much data that uses capital letters (mostly cursive handwriting).

Table 2. Testing the accuracy (ratio) of the model on the sample dataset.

Sample	EMNIST-LeNet5	IAM-CRNN	Tesseract
ijazah1.jpg	0.5	0.65	0.89
ijazah3.jpg	0.61	0.34	0.99
random3.jpg	0.40	0.48	0.97
random11.jpg	0.48	0.51	0.88
random12.jpg	0.55	0.52	0.94
Average	0.51	0.50	0.93

The test aims to find out how good the model is in reading handwritten numeric data. This makes the character recognition accuracy low depending on the segmentation results can be seen Fig. 10.

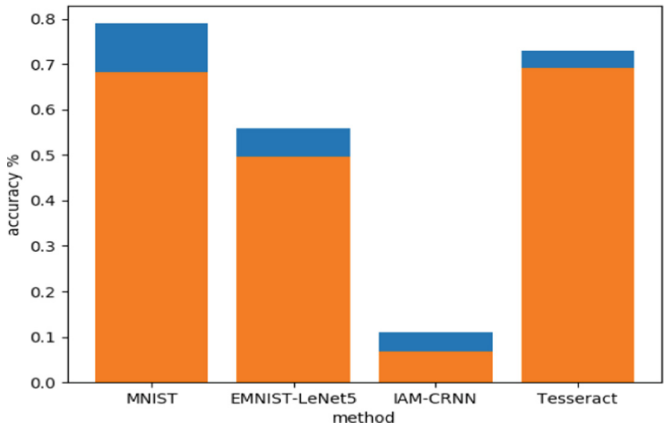

Fig. 10. Character recognition accuracy result

5 Conclusion

Based on the test results it can be concluded as follows:

- Run length smoothing algorithm (text) can be used to search for candidate data on diplomas. But it has a weakness in the accuracy when filtering the candidate.
- That dot_size configuration influential in segmentation, especially on images with different resolutions big. Larger image resolution makes the dots in the image larger, therefore it is necessary to standardize the process for automation.
- The minimum contour width parameter affects scanned images that have perspective. This parameter is useful as a filter for detected candidate lines.
- The accuracy of the RLSA segmentation method with point features has much higher accuracy than RLSA-TextFeature. The average accuracy obtained is 95.1% while RLSA-TextFeature only gets an average of 58.7%.

References

1. Supriana, I., Ramadhan, E.: Pengenalan Tulisan Tangan untuk Angka tanpa Pembelajaran. Konferensi Nasional Informatika, Bandung, Indonesia (2015)
2. Wirayuda, T.A.B., Syilvia, V., Retno, N.D.: Pengenalan Huruf Komputer Menggunakan Algoritma Berbasis chain code dan Algoritma sequence alignment, pp. 19–24. Konferensi Nasional Sistem dan Informat, Bali, Indonesia (2009)
3. Cohen, G., Afshar, S., Tapson, J., van Schaik, A.: EMNIST: extending MNIST to handwritten letters. In: International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, pp. 2921–2926 (2017). https://doi.org/10.1109/IJCNN.2017.7966217
4. Bunke, H., Marti, U.: The IAM-database: an English sentence database for offline handwriting recognition. IJDAR **5**, 39–46 (2002). https://doi.org/10.1007/s100320200071
5. SuperDataScienceTeam: Convolutional Neural Networks (CNN): Step 4—Full Connection (2018). Retrieved from superdatascience: https://www.superdatascience.com/blogs/convolutional-neural-networks-cnn-step-4-full-connection

6. Shi, B., Bai, X., Yao, C.: An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. IEEE Trans. Pattern Anal. Mach. Intell., pp. 2298–2304 (2016)
7. Borlepwar, A.P., Borakhade, S.R., Pradhan, B.: Run Length Smoothing Algorithm for Segmentation (2017)
8. Ujjwalkarn: An Intuitive Explanation of Convolutional Neural Networks, 29 May 2017 . Retrieved from ujjwalkarn: https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/
9. Setiawan, A., Hadi, I. P., Yoanita, D., Aritonang, A.I.: Virtual application technology of citizen journalism based on mobile user experience. J. Phys. Conf. Ser. **1502**(1), 012057. IOP Publishing
10. Setiawan, A., Rostianingsih, S., Widodo, T.R.: Augmented reality application for chemical bonding based on android. Int. J. Electr. Comput. Eng. **9**(1), 445 (2019)