A good CHoiCe: A Complex Handwritten Character dataset

Hongming Zhang[1], Josephine Plested[2], Sabrina Caldwell[2], and Tom Gedeon[2]

Research School of Computer Science,
Australian National University
hongming.zhang@anu.edu.au
jo.plested@anu.edu.au
sabrina.caldwell@anu.edu.au
tom@cs.anu.edu.au

Resumen The detection and recognition of cursive handwritten text is still a challenging problem. While much work has been done in this area and there are many mature and reliable commercial tools for neat and well structured handwriting, they are not as effective for more complex, changeable styles, and noisy character data sets. At present, in order to solve this type of problem, the lack of training data is problematic, especially for deep learning methods that require a large amount of training data. In this article, we provide the details of a newly created handwritten text dataset - CHoiCe. The data set consists of approximately 2,810 English and numeric characters composed of 62 standard English and numeric alphabet categories. The characters are in a similar style to the well know EMNIST dataset, but with more complex and noisy handwriting styles. In addition to presenting the data set, we compare CapsuleNet with Dynamic routing and CNN performance on CHoice and find that CapsuleNet is better suited to learning the more complex and noisy styles of handwriting in the dataset.

Keywords: Handwriting text· Dataset · CapsuleNet

1. Introduction

Single character recognition is an important step in the field of handwritten text recognition and detection [9]. An excellent handwritten character recognition and detection system will have a wide range of applications, such as handwriting recognition software, document translation software, and electronic handwriting documents[10]. At the same time, a high-performance handwritten character recognition and detection system can be easily combined with other text detection and text segmentation models to form a complete handwritten text detection and recognition system [9,11]. However, this task is not easy due to the complexity of style, shape, and format of handwritten characters [9].

Handwritten text datasets are the basis for an excellent handwriting detection and recognition system. When there are a sufficient number of standard

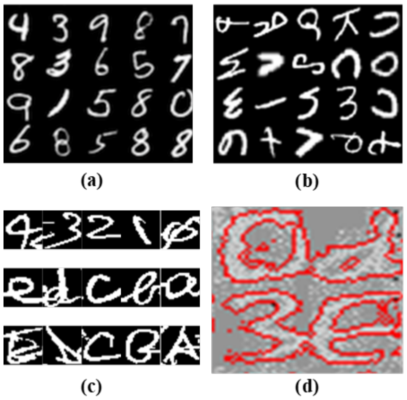


Figura 1: (a) Some examples of MNIST; (b) Some examples of EMNIST; (c) Examples of handwritten characters in CHoiCe; (d) Our capsule network tries to draw an accurate bounding box of handwritten characters

data sets with varying shapes, by using deep learning techniques, such as common CNN networks, one can easily get a good high-performance system; when the data is insufficient or too homogeneous, a system based on this handwritten text detection and recognition system obtained from this kind of data will not be versatile, and will be difficult to be detected and applied in practice.

The most widely used free handwritten character data sets are the MNIST and EMNIST data sets [1,3]. These data sets have tens of thousands of handwritten digits and letter data in a variety of styles; each class in these data sets has sufficient and balanced data quantity. While the datasets are large there is no great deviation between styles. This means they often don't reflect the diversity of actual writing in the wild, resulting in a handwriting recognition and detection system trained on the NIST series of data sets lacking versatility in practical applications.

In this work, we constructed a new open source small-scale handwritten character data set with a writing style that has many connected writings and background noise. This was created as part of an attempt to find a handwritten text detection and recognition method that can work well on complex cursive characters.

In addition, we compared the methods we used with the common KNN method and a simple CNN deep learning method, and conducted comparative tests on the dataset we constructed and the common NIST dataset, founded multiple baseline results of the dataset and proved that the capsule network has better adaptability to irregular writing.

2. Related work

2.1. Related Existing datasets

Many handwritten text data sets can be used for handwritten text detection and recognition, but there are not many free, complete, and practical data sets[7]. Here, we provide an overview of several representative English-word handwriting datasets.

The most famous and commonly used handwritten text data set is the National Institute of Standards and Technology Special Database 19 (NIST), which contains 3,600 different writers and approximately 800,000 tagged photos[6]. The NIST data set uses hand-printed Sample Forms to sample the numbers, English letters, and paragraph writing of each writer in a standard format [1,3].

The Modified NIST (MNIST) data set is currently the most convenient and famous baseline handwritten data set that can be used to train handwritten digit recognition and detection programs. The MNIST data set is often used as one of the benchmark tests in the field of image classification. It is used for 0-9 handwritten digit image classification tasks[2]. It contains 60,000 training images and 10,000 test images in 28 × 28 format[1]. MNIST data comes from the separate sampling of numbers in the NIST data set. It is a solution to the problem of streamlining the NIST dataset and the difficulty of using NIST data[1,3].

The Extended MNIST (EMNIST) data set is similar to the MNIST data set. It is one of the most widely used handwritten character data sets based on the NIST data set[7]. EMNIST contains about 800,000 independent handwritten digits and characters, covering digits 1 to 10, and a total of 62 categories in all upper and lower case of characters used in English [3]. EMNIST is an extension and supplement to the MNIST data set, making the source data of NIST easy to use, balancing and dividing the data set, and processing the images more clearly, making EMNIST more standard than MNIST[3].

The ÏAM"handwritten data set is a classic data set used in handwritten text detection, which contains 1,539 complete handwritten documents by 657 authors[8]. Because of the sufficient amount of data and large number of writers, the ÏAM"handwritten data set is one of the most widely used data sets today[4]. The ÏAM"data set includes three types of complete written pages, marked handwritten short sentences and marked handwritten words, which meets most of the application requirements [8] and is a very valuable free data set. However, because the data set is not pre-divided and does not provide the data type of a single handwritten character, the construction of a text recognition system based on character recognition is still lacking in usability.

The CEDAR data set is a well-known semi-open data set, including scene text, printing, handwriting, and other text data in various states, and it is free to everyone with thousands of unlabeled sheets written by roughly 1,500 different writers handwritten text [5]. For handwritten text data, the CEDAR dataset provides a large amount of clear data in four cases of complete handwritten text, sentences, words, and letters that are labelled but an access fee is charged.

It has the most writing styles [5], which are suitable for most situations. However, the free parts are difficult to use because they are not marked, and the processing and application of data is too difficult.

3. Data set description

Cuadro 1: CHoiCe Dataset description

Data Sources	Number of photos	Percentage of CHoiCe
Informal meeting notes	2196	77.2 %
CEDAR	614	21.8 %

The CHoiCe dataset [1] is composed of the daily meeting records of an university academic team with some additional data coming from the CEDAR dataset to balance the number of samples in each category of character. The data consists of complex writing styles, shorthand and non-standard shapes and sizes of letters. The number of samples in each category of characters is not less than 40. The writing of the original data mainly uses black and blue pens, but at the same time has different shades of writing. Because the data is collected from daily notes, the data has most of the situations that will appear in the everyday use.

The meeting notes are recorded on standard unlined A4 paper, and are converted into a complete full-page handwritten note record using high-definition scanning. The handwritten data does not have any format and regional requirements, thus achieving the most authentic data source to the greatest extent.

The meeting notes, mainly consists of the following parts: The first line is the name of the current meeting participant and the meeting date. This line ensures sufficient data about capital letters and numbers. The following part is the subject content, mainly including more formal conversation records, mathematical drafts and diagrams, border inserts and comments, and correction of errors.

It should be noted that because the volume of original data is not large, and the number of writers is small, the data in the data set is biased. At the same time, because the data comes from real-time meeting records, many writings are not formal word spellings, which are personal shorthand methods and it is difficult to construct a traditional corpus. For example, "2ïs often used instead of "toïn notes. In addition, in order to write quickly, the way of writing is not clear ligature and cursive, so some character data is difficult to use with algorithms to achieve accurate segmentation, and manual collection and calibration are required.

[1] The CHoiCe Dataset is open source at ANU Data Commons: https://dx.doi.org/10.25911/602355a95f787

The data set contains a total of 2810 character-level data pictures, divided into 10 types of handwritten digital characters, "0"to "9"; 26 types of handwritten lowercase English letters, 'a' to 'z'; 26 types of handwritten uppercase English letters, 'A' to 'Z'. The data set contains a total of 62 categories.

Each picture in the data set is saved in the same way as MNIST[2], in the "PNGïmage format of 28 × 28. The data set contains 62 categories of raw data that are not binarized and 62 categories of binarized data that are exactly the same as the MNIST format, which is convenient for direct use and researchers who need to try more advanced image preprocessing methods.

4. Method

4.1. Exploration of possible application methods on cursive and irregular writing characters

Type	Configurations
Softmax	-
MaxPooling	Window:2×2
Conv2D	maps:64, k:3×3
MaxPooling	Window:2×2
Conv2D	maps:32, k:3×3
Input	28 × 28 × 1

Figura 2: Simple CNN structure

Capsule-Net based on dynamic routing algorithm has excellent and comprehensive feature synthesis ability and image reconstruction ability [12]. The features that Capsule-Net can apply are no longer limited to edges, corners, colors, etc., but also can extract vector features such as image position and direction, which makes Capsule-Net able to obtain more information and improve accuracy when identifying objects with consistent content but changeable style, rotation, location of features etc.

In the capsule network, recognition of objects takes place by the network decoding the features that have been obtained through the capsule network and reconstructing the corresponding object. When the result of a certain class of reconstruction is most similar to the actual object, the object will most likely belong to this category. In this process, the reconfiguration ability of the capsule network is excellent, so in this project, we decided to use the capsule network to attempt to generate more clear and understandable handwritten text for scribbled and illegible handwritten text.

As a classical network structure of multi-layer feature fusion, U-net [13] is often used to enhance the utilization effect of convolution networks for multi-level features. In the previous layer, the PrimaryCaps part of capsule-net was

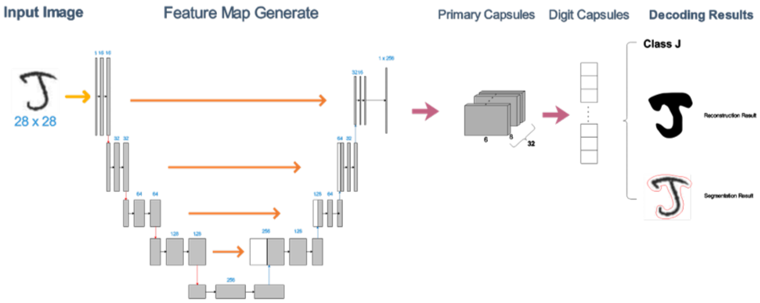


Figura 3: UNet-CapsuleNet structure

designed as a U-net structure. To the best of our knowledge no one has previously implemneted a capsule network in this way.

In order to get a clearer result of the reconstructed image, so that the model can meet the project's expected ability to make the chaotic or defective handwritten text clearer and easier to understand, we use standard handwriting (similar to MNIST and clear handwriting) as the object of feature extraction, so as to make the features stored in the final çapsuleçlearer.

The U-Net network is suitable for multi-scale feature extraction. In the case of less training data, compared with using simple convolution network for feature extraction, U-Net is able to extract features in different scales for a single image, avoiding the problem of only extracting the features of the external outline of the object and neglecting the internal detail features. The original capsule network uses a simple convolution layer to extract the input image into a feature map, which has a single scale and may cause important information loss, such as scribbled handwritten text because of the uneven thickness of pen and ink. It is necessary to carry out feature extraction on a smaller scale to get the relevant information. Therefore, in order to have more comprehensive feature extraction results in the feature extraction stage when using the capsule network to generate a clear handwritten image, we creatively combine U-net with the traditional capsule network and use U-Net to replace the feature extraction part of the original capsule network, so as to get a more detailed feature map.

In the original capsule network, reconstruction is a key step to get the classification to which the picture belongs, and the category is determined by comparing the similarity between the reconstructed results and the original image. In an innovative approach, we use the reconstructed results as the desired results and the mask used for segmentation when applying the capsule network. Fig.3. shows the structure of the network that we designed to combine with U-Net. The network structure is clearly divided into two parts. The left half uses the classic U-Net architecture; it consists of a contracting path and an expansive path. At the end of Figure 5 is an example of the output of the network, with the letter

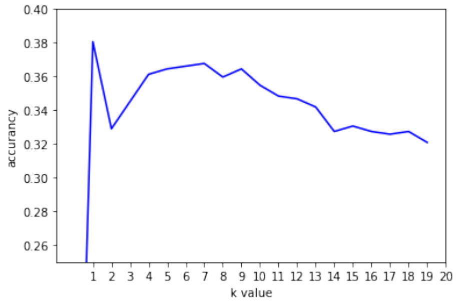


Figura 4: k neighbours value choosing

"J.ªs an example. The result of the reconstruction of the capsule network can be used as a çlearer handwriting". At the same time, the reconstruction result can also be used as a mask to accurately segment the handwritten letters in the original image. Finally, the category corresponding to the current handwritten letter is also output.

5. Results and discussion

5.1. Classification Baselines fo CHoiCE

Cuadro 2: Test set accuracy(the mean of each class's accuracy to avoid class imbalance)

Model	MNIST	CHoiCe
1-Nearest Neighbour Baseline(KNN)[15]	99 %	38 %
Simple CNN	98 %	54 %
Original Capsule-Net	99 %	83 %

Table 1 shows the baseline classification results of Mnist and Choice, and MNIST used to verify that the model works smoothly[14]. We consider 3 different baselines: a simple 1 - neighboring neighbor algorithm[15], a Simple CNN and an Original Capsule-Net. The neighbor value of the KNN baseline is based on a large number of tests in Fig.4. taking the maximum value.

5.2. Comparing Unet-Capsule-net with Simple CNN

In order to finally understand the actual effect of the Unet-Capsule-net model an appropriate Loss function is needed. We continue to use the Margin loss [12]

Cuadro 3: Simple CNN test result

Model	Training Dataset	Test Dataset	Accurancy
Simple CNN	MNIST	MNIST	98 %
Simple CNN	Original CHoiCe	Original CHoiCe	54 %
Simple CNN	Original CHoiCe	Reconstructed CHoiCe	51 %
Simple CNN	Reconstructed CHoiCe	Original CHoiCe	47 %
Simple CNN	Reconstructed CHoiCe	Reconstructed CHoiCe	19 %

Cuadro 4: Original Capsule-Net test result

Model	Training Dataset	Test Dataset	Accurancy
Original Capsule-Net	MNIST	MNIST	99 %
Original Capsule-Net	Original CHoiCe	Original CHoiCe	83 %
Original Capsule-Net	Original CHoiCe	Reconstructed CHoiCe	77 %
Original Capsule-Net	Reconstructed CHoiCe	Original CHoiCe	69 %
Original Capsule-Net	Reconstructed CHoiCe	Reconstructed CHoiCe	23 %

Cuadro 5: Blank data add-in test result

Model	Training Dataset	Test Dataset	Accurancy
Original Capsule-Net	Original CHoiCe + 15 % blank pictures	Reconstructed data	14 %
Simple CNN	Original CHoiCe + 15 % blank pictures	Reconstructed data	11 %

in the original capsule network paper as the loss function in model training and testing.

The mathematical expression of Margin loss is as follows($\lambda = 0{,}5$)[12] :

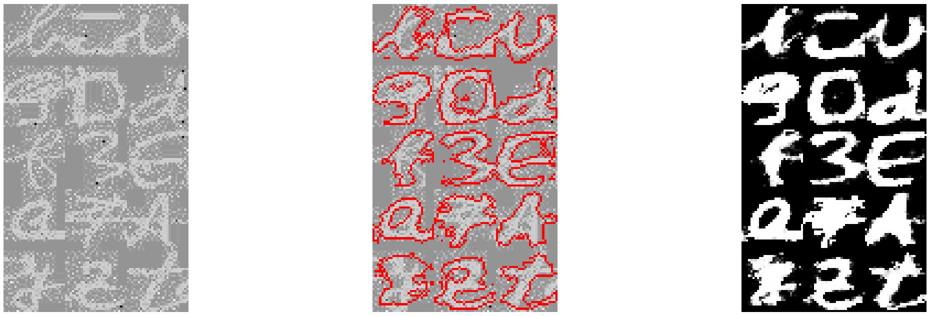
$$L_k = T_k max(0, m^+ - ||vk||)^2 + \lambda(1 - T_k)max(0, ||vk|| - m^-)^2 \tag{1}$$

The length of the vector used in the capsule network represents the probability of the existence of the entity, so in the training process of the model, we anticipate that when there is a corresponding entity in the picture, the corresponding Digit capsule will have a larger vector length.

The output of the DigitCaps layer has 62 16-dimensional vectors. During training, for each training sample, the loss value of each vector is calculated according to the above formula, and then 62 loss values are added to get the final loss. For supervised learning, each training sample has the correct label, in which case it will be a 62-dimensional one-shot vector consisting of 61 zeros and one (correct position). In the loss function formula, the correct label determines the value of T_k; if the correct label corresponds to the number of a particular DigitCap, T_k is 1, otherwise it is 0.

For example, suppose the correct label is 1, which means that the first DigitCap is responsible for encoding the existence of the number 1. The T_k of the loss function of this DigitCap is 1, and the T_k of the remaining 61 DigitCaps is 0. When T_k is 1, the second term of the loss function is zero, and the value of the loss function is calculated by the first term. In our example, to calculate the loss of the first DigitCap, we subtract the output vector of this DigitCap from m^+, where m^+ takes a fixed value of 0.9. Next, we keep the obtained value and square it. Otherwise, 0 is returned. In other words, when the probability of the correct DigitCap predicting the correct label is greater than 0.9, the loss function is zero, and when the probability is less than 0.9, the loss function is not zero.

For DigitCap that does not match the correct label, T_k is zero, so the second term is calculated. In this case, when the probability of DigitCap predicting incorrect label is less than 0.1, the loss function is zero, and when the probability of predicting incorrect label is greater than 0.1, the loss function is not zero.


(a) Original data with noise

(b) Detection results

(c) Reconstruction results

Figura 5: Results for my capsule-net working

Fig.5. shows the detection and reconstruction results of the model on an input image with random white noise. Fig.5. (a) is an input image with random noise added. The 15 random test cases are 'l', 'I', 'v', '9', 'D', 'd', 'f', '3', 'E', 'a', '4', 'A', 'y', 'z' and 't'. Fig.5.(b) shows the results after testing using the Unet-Capsule-net designed in this project: almost all targets are depicted with more accurate arbitrary shape bounding boxes, eliminating the problem of noise interference when detecting and cutting these detected handwritten texts, and achieving the original purpose of the project. Fig.5. (c) shows the test input for (a), using Unet-Capsule-net to generate clear, noise-free handwritten results. It can be found that after the reconstruction of the original handwriting data using Unet-Capsule-net, the noise added in (a) is first avoided, which sets the text in a purer environment, which is conducive to the text recognition task. Secondly, the reconstructed image makes the extra strokes brought by some consecutive strokes or cursive script omitted, such as "fin the picture. If the result is clear enough, the project will fully achieve the goal of "getting specific, clear and complete handwritten text detection results".

In order to know whether the reconstructed image is clear enough and easy to identify, We chose to use the Unet-Capsule-net model to completely transform the dataset into the reconstructed image composition, and use this dataset for a series of tests. We use a simple CNN model (composed of two layers of convolution layer, one layer of max-pooling and one layer of softmax) and the original capsule network in this paper for comparative training and testing. If the performance of the model is not greatly degraded or improved when the reconstructed results are used as data in the training set or test set, it means that the picture obtained by the refactoring is sufficiently clear. For each model, there are a total of five sets of tests. The first group is the test on MNIST, which is used to verify the availability of the model. The second group of experiments is to train on .$^{\mathrm{O}}$riginal CHoiCe.$^{\mathrm{a}}$nd test on .$^{\mathrm{O}}$riginal CHoiCe"to get the baseline for complex handwriting problems. The third group of experiments is to train on .$^{\mathrm{O}}$riginal CHoiCe.$^{\mathrm{a}}$nd test on Reconstructed CHoiCe"to verify whether the reconstructed image can be clearly recognized when the model obtains correct knowledge. The fourth group of experiments are trained on Reconstructed CHoiCe.$^{\mathrm{a}}$nd tested on .$^{\mathrm{O}}$riginal CHoiCe"to verify whether the reconstructed images have enough and clear features, that is, the model can recognize the actual handwritten text. The fifth group of experiments are trained on Reconstructed CHoiCe.$^{\mathrm{a}}$nd tested on Reconstructed CHoiCe"to test whether the reconstructed images can completely replace the actual handwritten results.

Table 3. shows the results of five sets of tests for the simple CNN model, and Table 4. shows the results of five sets of tests for the Original Capsule-Net model. When Capsule-Net and simple CNN have the same recognition performance for MNIST, it can be found that using Capsule-Net has better performance than CNN in recognizing complex handwritten text. This is because Capsule-Net can extract more information such as the direction and location of features when facing cursive or handwritten text with changeable style, while simple CNN can only obtain simple image information. As a result, Capsule-Net has better performance in the face of complex handwritten text situations.

In addition, comparing the results of four groups of experiments on Reconstructed CHoiCe.$^{\mathrm{a}}$nd tested on .$^{\mathrm{O}}$riginal CHoiCe", it can be found that when using .$^{\mathrm{O}}$riginal CHoiCeïn the training set, whether using the original test set of .$^{\mathrm{O}}$riginal CHoiCe.$^{\mathrm{o}}$r using the test set composed of reconstructed images in Reconstructed CHoiCe", the performance changes of simple CNN and Original Capsule-Net are very small. This proves that after the complete features are extracted, the model can extract the necessary features from the reconstructed image of the test set for recognition, and the reconstructed image can be recognized, and the result is similar to that of the original image. This indicates that when the "knowledgeïs rich and accurate enough, the reconstructed image is easy to identify.

Finally, the performance of both models was greatly degraded when both training and testing were in the case of Reconstructed CHoiCeïn both Table 3.

and Table 4. This is because some of the reconstructed images are very bad and can not replace the original handwritten image, and there is a lot of feature loss and confusion, which makes the knowledge obtained by the model "wrong". In order to verify whether useless and incorrect reconstructed images can cause this phenomenon, as shown in Table 5. we randomly replaced 15 % of .$^{\text{O}}$riginal CHoiCe"with completely blank images, and obtained a model performance similar to that of the previous model. It is proved that both a higher proportion of errors or blank data will impact negatively on the performance of the model.

6. Conclusion

This work establishes a data set containing handwritten Arabic numerals and English cursive handwriting. The data is collected from handwritten texts in daily situations. The data set can be used as a supplement to the benchmark data set EMNIST to develop and evaluate handwritten text recognition systems and handwritten text detection systems, making text recognition and detection more useful for situations where unstructured handwriting is used. The data set will be continuously expanded and improved by continuing to collect data so that it can be used more widely.

In addition, this work also uses simple KNN, simple CNN and UNet-CapsuleNet to benchmark the dataset and explore and try handwriting recognition and detection tasks in complex environments to confirm the complexity and usability of the data set. It was discovered that the possible solution for obtaining more accurate text recognition and detection results is to take more features into consideration, based on the result that simple CNN and our Capsulenet have similar high performance on MNIST but Capsulenet has significantly higher performance than Simple CNN on our CHoiCe' complex data, as 83 % accuracy for Capsulenet and 54 % for Simple CNN. We conclude that the end-to-end handwriting recognition model with Capsulenet can better pass the angle and direction of the text. This allows for better recognition of more complex handwriting styles.

Referencias

1. Yann Le Cun, Corinna Cortes, and Christopher J. C. Burges. The MNIST database of handwritten digits. http://yann.lecun.com/exdb/mnist/, 1994. MNIST was created in 1994 and released in 1998.
2. Yadav, C. and Bottou, L., 1905. Cold Case: The Lost MNIST Digits. 2019.
3. Cohen, Gregory, Saeed Afshar, Jonathan Tapson, and Andre Van Schaik. .$^{\text{EM}}$NIST: Extending MNIST to handwritten letters.Ïn 2017 International Joint Conference on Neural Networks (IJCNN), pp. 2921-2926. IEEE, 2017.
4. U.-V. Marti and H. Bunke. A Full English SentenceDatabase for Off-Line Handwriting Recognition. InProceedings of the Fifth International Conference onDocument Analysis and Recognition, ICDAR '99, pages705–, Washington, DC, USA, 1999. IEEE ComputerSociety.
5. S. Srihari, S.-H. Cha, H. Arora, and S. Lee. Individu-ality of handwriting: a validation study. InDocumentAnalysis and Recognition, 2001. Proceedings. Sixth International Conference on, pages 106 –109, 2001.

6. Founds, Andrew P., Nick Orlans, Whiddon Genevieve, and Craig I. Watson. "Nist special databse 32-multiple encounter dataset ii (meds-ii)."(2011).
7. Al Maadeed, Somaya, Wael Ayouby, Abdelaali Hassaine, and Jihad Mohamad Aljaam. "QUWI: an Arabic and English handwriting dataset for offline writer identification.Ïn 2012 International Conference on Frontiers in Handwriting Recognition, pp. 746-751. IEEE, 2012.
8. M. Liwicki and H. Bunke. IAM-OnDB - an On-LineEnglish Sentence Database Acquired from HandwrittenText on a Whiteboard. InProceedings of the EighthInternational Conference on Document Analysis andRecognition, ICDAR '05, pages 956–961, Washington.
9. Beigi, Homayoon SM. .An overview of handwriting recognition.Ïn Proceedings of the 1st Annual Conference on Technological Advancements in Developing Countries, Columbia University, New York, pp. 30-46. 1993.
10. Memon, Jamshed, Maira Sami, Rizwan Ahmed Khan, and Mueen Uddin. "Handwritten optical character recognition (OCR): A comprehensive systematic literature review (SLR).ÏEEE Access 8 (2020): 142642-142668.
11. Carbonell, Manuel, Joan Mas, Mauricio Villegas, Alicia Fornés, and Josep Lladós. .End-to-end handwritten text detection and transcription in full pages.Ïn 2019 International Conference on Document Analysis and Recognition Workshops (ICDARW), vol. 5, pp. 29-34. IEEE, 2019.
12. Sabour, Sara, Nicholas Frosst, and Geoffrey E. Hinton. "Dynamic routing between capsules..arXiv preprint arXiv:1710.09829 (2017).
13. Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. Ü-net: Convolutional networks for biomedical image segmentation.Ïn International Conference on Medical image computing and computer-assisted intervention, pp. 234-241. Springer, Cham, 2015.
14. Clanuwat, Tarin, et al. "Deep learning for classical japanese literature..arXiv preprint arXiv:1812.01718 (2018).
15. Guo, Gongde, et al. "KNN model-based approach in classification..OTM Confederated International Conferences.On the Move to Meaningful Internet Systems". Springer, Berlin, Heidelberg, 2003.