Preparation of Papers for IEEE TRANSACTIONS and JOURNALS

First A. Author, Second B. Author, Jr., and Third C. Author, *Member, IEEE*

***Abstract*—These instructions give you guidelines for preparing papers for IEEE TRANSACTIONS and JOURNALS. Use this document as a template if you are using Microsoft *Word* 6.0 or later. Otherwise, use this document as an instruction set. The electronic file of your paper will be formatted further at IEEE. Define all symbols used in the abstract. Do not cite references in the abstract. Do not delete the blank line immediately above the abstract; it sets the footnote at the bottom of this column.**

***Index Terms*—About four key words or phrases in alphabetical order, separated by commas. For a list of suggested keywords, send a blank e-mail to keywords@ieee.org or visit .http://www.ieee.org/documents/taxonomy_v101.pdf**

I. INTRODUCTION

Advance in digital health have improved the quality of health service for individuals and public. For example, physicians can track patient demographics, medication, and laboratory tests and read and store clinician notes through Electric Health Record system (EHR) in United States. Access to integrated patients' records can enhance health outcomes by improving medical diagnosis, data-based treatment decisions, digital therapeutics, clinical trials, self-management of care and person-centered care as well as creating more evidence-based knowledge, skills and competence for professionals to support health care. Some developed countries have made the considerable progress in digital healthcare, however, a person's medical records still consist of a few sheets of paper in low-and middle-income countries (LMICs) because digitalized health system like EHR requires a significant financing and involvement of various stakeholders [16], [17]. In addition, percentage of physicians adopted EMR/HER is less than 90% in U.S. (Percent of office-based physicians using any EMR/EHR system: 89.9%, Percent of office-based physicians with a certified EMR/EHR system: 72.3%) [30]. It means that many doctors still use paper and pen to write prescription.

On the other hand, text recognition techniques have been developed for handwriting [18] – [27]. In addition, application of text recognition in healthcare has been studied in some research. [28] studied textual information extraction from images of medical laboratory reports using deep learning approach, and [29] suggested online handwriting recognition system to predict doctor's handwritten prescription.

Considering that paper-based medical and health records are still dominating in many countries, this paper presents hand-written text recognition system for medical records forms. The system employed detection of elements in medical forms such as text box, checkbox and obtained character-level images through image processing. Features of letter or digit are extracted and classified from these character-level images using a convolutional neural network (CNN). Checkboxes are also classified as 'yes' or 'no' depending on the number of black pixels. This recognition system is evaluated on 125 hand-written documents collected from Moldova for Tuberculosis Research.

This paper is organized as follows. In Section II, previous research on text recognition and application in healthcare was investigated. Section III introduce the designed pipeline for handwriting recognition. In Section IV, the process of building dataset is described, and CNN structure, parameters, training and evaluation of classification are explained in Section V. the results of experiments and performances are presented in Section VI, and Conclusion is discussed in Section VII.

Manuscript received October 9, 2014. (Write the date on which you submitted your paper for review.) This work was supported in part by the U.S. Department of Commerce under Grant BS123456 (sponsor and financial support acknowledgment goes here). Paper titles should be written in uppercase and lowercase letters, not all uppercase. Avoid writing long formulas with subscripts in the title; short formulas that identify the elements are fine (e.g., "Nd–Fe–B"). Do not write "(Invited)" in the title. Full names of authors are preferred in the author field, but are not required. Put a space between authors' initials.

F. A. Author is with the National Institute of Standards and Technology, Boulder, CO 80305 USA (corresponding author to provide phone: 303-555-5555; fax: 303-555-5555; e-mail: author@ boulder.nist.gov).

S. B. Author, Jr., was with Rice University, Houston, TX 77005 USA. He is now with the Department of Physics, Colorado State University, Fort Collins, CO 80523 USA (e-mail: author@lamar. colostate.edu).

T. C. Author is with the Electrical Engineering Department, University of Colorado, Boulder, CO 80309 USA, on leave from the National Research Institute for Metals, Tsukuba, Japan (e-mail: author@nrim.go.jp).

II. LITERATURE REVIEW

A. Optical character Recognition

Optical Character Recognition or optical character reader is a technology that transforms images of typed, handwritten or printed text into machine-translated text. Optical Character Recognition enables an extraction of the relevant information and data entry into a database instead of retyping the text manually. Optical Character Recognition can be applied in various fields such as banking, legal industry, digital libraries, optical music recognition, automatic number plate recognition, and handwritten recognition [1]. For example, in banking, OCR is widely used to scan the amount of the issued checks and transfer the correct amount of money. In addition, the application of OCR can provide open access literature by reading scanned documents. Information in scanned papers or books can be obtained and retained in digitized form. Automatic number plate recognition is to identify vehicle registration plates by utilizing optical character recognition. This technique is vastly employed by policy forces and as a way of electronic toll collection and recognizing the movement of traffic. A variety of OCR engines are available now. Tesseract, open-source OCR engine, is the one of well-known OCR tool. The engine was developed at HP between 1984 and 1994 and kept developed by google from 2006 [2]. The latest stable version is Tesseract 4.1.1, which can recognize more than 100 languages and are based on deep learning-based model with LSTM(Long Short Term Memory). In google,

Tesseract is utilized for text detection on mobile devices, in video, and in Gmail image spam detection. Any other OCR engines including Google Cloud Vision, ABBYY FineReader, OCRopus are commercial engines. These OCR engines provide high accuracy in character and word recognition. For example, Tesseract 4.0(based on LSTM) has shown 7.25% - 16.97% Character Error Rates and 11.41% - 40.43% word error rates depending on languages [3]. Despite high accuracy in character and word recognition, the use of the existent OCR engines is expensive, and have a limitation in applying the engines directly to target data. This is because OCR engines are trained from easily recognizable and machine-readable number or text such as road sign,

B. *Handwriting Recognition*

OCR technology has been developed since the 70s and shown 99% accuracy with typed character in high-quality images. However, handwriting recognition has still been a challenge in traditional OCR because of diverse style of writing and cursive handwriting. Handwriting Recognition(HWR), also known as Handwritten Text Recognition(HTR), is a technology that a computer can receive and interpret handwritten text from paper documents, photographs, touch-screens and other devices. Handwritten text images includes paper documents by optical scanning(off-line sensed) and pen tip's movement on the screen of devices (online sensed). In this paper, offline handwriting will be focused because

Offline handwriting systems generally consist of four steps: acquisition, segmentation, recognition, and postprocessing. First, the handwriting to be recognized is digitized through scanners or cameras. Second, the image of the document is segmented into lines, words, and individual characters. Third, each character is recognized using OCR techniques. Finally, errors are corrected using lexicons or spelling checkers. Traditional optical character recognition had been focused on extracting text from scanned document or photos. Text extraction in natural scene has been more challenging because of scene complexity including heterogeneous background, noise, distortions, and directional blur. To deal with these difficulties, deep learning based model has been developed for handwriting recognition.

The first handwriting recognition(HWR) models were based on hidden Markov models[4], and SVM [5]. Rosetta uses ResNet as a feature extractor and CTC-loss as a character predictor [6].

According to Alejandro et al [10], there has been a lot of progress in handwritten character recognition with the advent of deep learning techniques. In the survey, many researchers investigated and compared the performance of convolutional neural networks(CNNs) and other deep learning algorithms with MNIST and EMNIST datasets (MNIST is a basic dataset for a standard benchmark in classification and computer vision. Each image in MNIST has a corresponding label, a number between 0 and 9 representing the digit drawn in the image. EMNIST dataset is an extended version of MNIST dataset, which includes handwritten digits and alphabetic characters. EMNIST images have also 28x28 pixel image format and dataset structure that directly matches the MNIST dataset. Both MNIST and EMNIST datasets are subsets of a much larger dataset called NIST Special Database 19) [11]. Especially, a test error rate was compared between algorithms and the study showed that the top state-of-the-art work(based on CNNs) gained 0.21% test error rate with MNIST. With EMNIST dataset, the highest test accuracies were 99.79% and 95.44% for digit and letter respectively.

C. *Handwriting Recognition in Healthcare*

Paul et al [7] has tested the performance of OCR software on encounter forms in clinics. Encounter forms have some fields which are written by nurses or physicians to measure the vital signs and other numeric observations. A commercial OCR engine called 'Teleforms Elite 7.1' were used for handwriting recognition, and digit recognition rate was 92.4% with 95% confidence. Luke et al[8] proposed an optical character recognition pipeline using the existent third-party OCR engines. The aim of this research was to identify cataract type and severity. Random sample images of 949 forms were collected and fields related to cataracts were identified by an expert. After extracting the fields from image documents and pre-processing, three OCR engine(Tesseract, LEADTOLLS ICR module, Nuance OmniPage) were used to recognize the hand-written text. Finally, postprocessing were applied by using the context of surrounding character and results from known misclassification in order to minimize the false-negative, false-positive, and character-substitution errors from results of three engines. As the result, the pipeline showed a positive predictive value of 94.6% and a sensitivity of 13.5%. In another study, a multi-modal strategy with combinational use of database query, natural language processing(NLP) and optical character recognition(OCR) were proposed to identify cataract subjects and related attributes from electronic health records(EHRs) including free-text documents and scanned clinical images. Positive predictive values (PPVs) from multi-modal strategy surpassed 95% [9].

III. Pipeline Design

Figure.1. describes the whole process to read a document automatically, and details can be seen below.

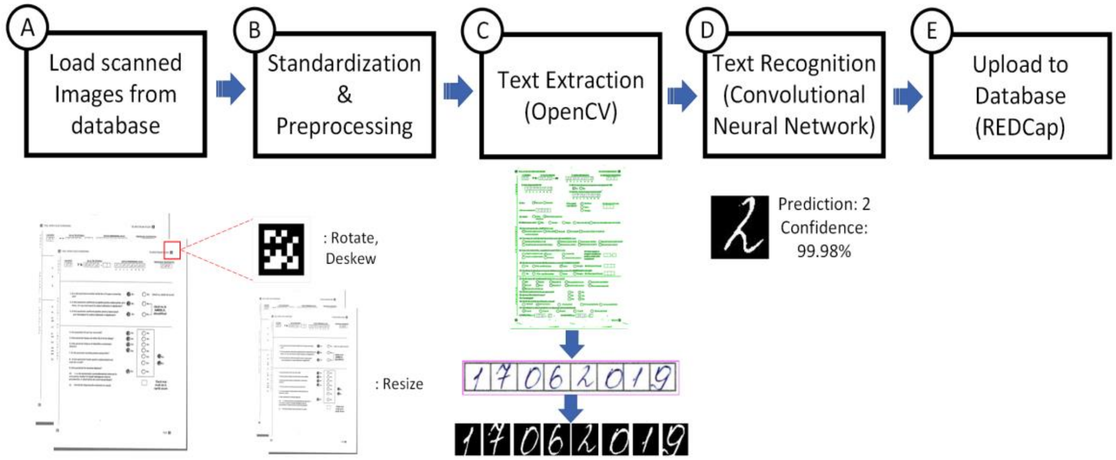


Fig.1. Pipeline for reading medical forms automatically

A. *Scan files from Moldova*

Total six types of forms were all collected from Moldova. (how many forms and cases were collected?) Every form has an identifier printed on the four corners of the document to assist with tracking form templates and deskewing the form properly. The identifier is called AprilTag, which are a type of fiducial marker consisting of a black square with a white foreground that has been generated in a particular pattern. All forms were written by nurses in Moldova manually. Written documents are scanned and then stored in a secured database.

B. *Standardization & Preprocessing*

Second step is standardization and preprocessing the forms. While scanning documents in Moldova, scanned file can be skewed or flipped upside down. In addition, there is a resolution differences between scanned image files depending on scanner. These two factors can cause lower performance in character recognition. Four AprilTags printed on each form are utilized to deskew and rotate the document in the right manner, and then scanned images were resized using OpenCV(version 4.5.3) from Python in order to obtain a same-sized and resolution images.

C. *Detection*

Each form consists of text boxes and check boxes, and text boxes were considered as bounding boxes to extract all data fields. Using approxPolyDP() function in OpenCV, the coordinates and size of bounding boxes were determined. Detected bounding boxes were cut into character-based small boxes(Only one letter or digit is in the small box). These small boxes were again processed to remove noises and bordering lines of the box which can lower the performance of digit and letter recognition. For detection of check boxes, HoughCircles() function in OpenCV were used. To clean up checkbox images, boundaries of circles were eliminated and then to figure out the check boxes (marked as green circles in figure 1.2) were marked or not, total number of black pixels were counted in the region of interest (ROI). If the number of black pixels in check box exceeds a certain number, the checkbox is recorded as ‘marked’, otherwise ‘not marked’.

D. *Letter & digit recognition*

Each letter or digit image in small box is recognized from pre-trained classifier. The classifier is based on Convolution Neural Network (CNN) trained letter and digit images from Moldova data. For letter recognition, EMNIST dataset was also trained with Moldova letter data because of lack of letter images from scanned files. Prediction label of the input image will be presented with confidence of the prediction in order to review the result before entering the data into database.

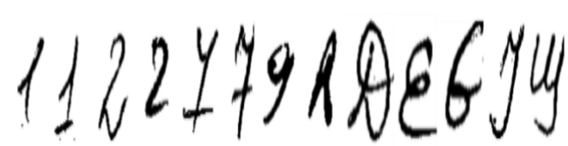


Fig.2. Examples of handwritten digits and letters from Moldova (number 1, 1, 2, 2, 7, 7, 9, and alphabet A, D, E, G, J, Y)

E. *Upload to Database*

By coordinates of small boxes and checkboxes, all predicted letters and digits, and marked circles are re-organized and reconstructed as one record. One record indicates one scanned file and the digitized scanned image enters into REDCap database.

IV. Dataset

Based on the previous studies, Convolutional Neural Networks(CNNs) was implemented on this study. For the initial trial, EMNIST dataset was used for handwriting recognition for medical forms from Moldova. However, recognition rate for letter and digits were less than 80% test accuracy with EMNIST dataset.

Analysis for the low recognition revealed several causes of the poor performance. Main reason is that nurses’ writing style in Moldova is totally different from the writing style of EMNIST dataset (see fig.2). Despite of common alphabets and numeral systems in the world, styles of handwritten letter and

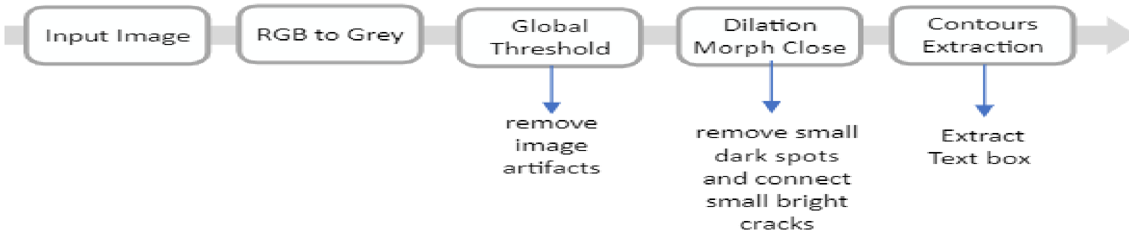


Fig.3. Image processing for text extraction

Fig.4. Effects of image processing techniques

digits vary between individuals, and even regions. For example, the set of images in the MNIST and EMNIST consist of digits written by U.S. high school students and employees of the United States Census Bureau. Meanwhile, collected medical forms were written by nurses in Moldova. Especially, for some specific digits and letters, there were prominent variation between U.S. and Moldova's Writing.

In order to learn Moldova's writing style, new hand-written dataset was created from collected forms in Moldova. Image processing process was illustrated as below in Fig.3.

A. Image Processing

Six type of medical forms to collect data on Tuberculosis(TB) patients and family members has been documented by nurses in Moldova. Six type forms are: TS01, TS02A, TS02B, TS03, TS04, TS05A, TS05B, and TS06. All forms have written by hands, been scanned and stored in a database. From the database, scanned image files were loaded and boxes containing letter or digit were detected using OpenCV. All detected boxes broke down into character-level boxes(Fig.6). In more details, an input image in RGB is transformed to gray-scale image. For the gray image, thresholding is applied to remove noises on the image. In this research, three kind of thresholding techniques- Adapted-Gaussian, Adapted-Mean, Global- were tested. The result of thresholding is shown in Fig. 5. After thresholding, dilation followed by an erosion(defined as morph close) was applied to eliminate pepper noise and connect small bright cracks. In Fig.4 (c), you can see the effect of dilation which enlarge the boundaries of foreground(white) regions in an image. After then contours were extracted for text boxes on the form and some margins of 5 pixels were added to obtain text boxes in Fig.4(d). Checkboxes(circle-shaped) were also detected with HoughCircle function in OpenCV in the similar way for text boxes. The only difference is that 'Morph Close' step was skipped for circle detection in an image. Extracted text boxes were cut into small boxes containing a character or digit by finding inner contours and rectangles in text boxes.

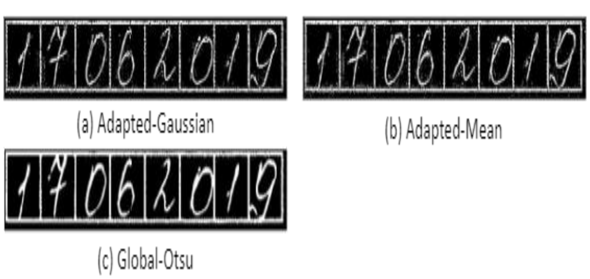


Fig.5. Effects of various thresholding methods

Fig.6. Extracted digit or letter images (28 x 28) after processing

B. Dataset Structure

All processed character-level images were collected into a dataset. The dataset is split into three parts: 8,000 digit images of training data, 2,000 digit images of validation data, and 1,000 digit image test data. For letter, there are 2,000 training data points, 500 training data points, and 125 test data points. All images were collected from TS01, TS02 and TS03. Each image is a 28 x 28 binary image and has a label corresponding to digit(0-9) or letter(A-Z). While making up the dataset, imbalance between classes were observed. For example, the number of collected image of 0,1,2,8, and 9 were big because every patient has been required to tell his or her age, and birth year for every form. On the contrary, number of 3,4,5,6, and 7 consisted minor data points of digit dataset. Imbalanced datasets are those where there is a severe skew in the class distribution and lead learning algorithms to ignore the minority class entirely. This imbalance can cause the poor predictive performance in classification, specifically for the minority class. One method for imbalanced dataset is to randomly resample the dataset. In this research, oversampling – randomly duplicates examples from the minority class in training dataset, was applied. For instance, 8,000 digit training datapoints are the sum of 800 data points of each number class. In addition, for learning in alphabet characters, EMNIST data were combined with collected letter dataset. In the medical form, letter can be found only for Nurse Initial field and it caused sparsity of datapoint in alphabet classes. In order to acquire enough data for learning, EMNIST dataset were utilized with our own letter

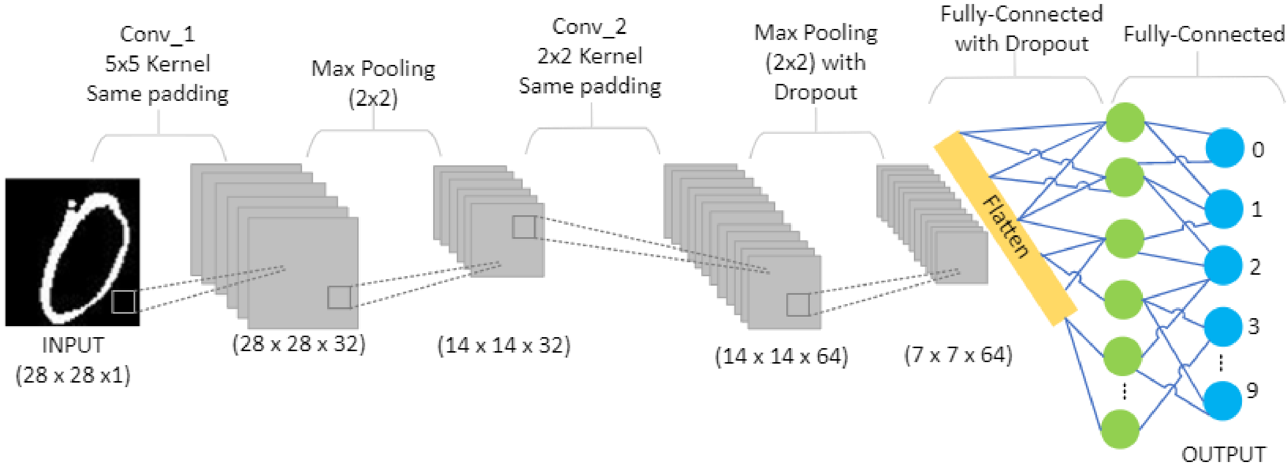


Fig.7. Structure of Convolutional Neural Networks

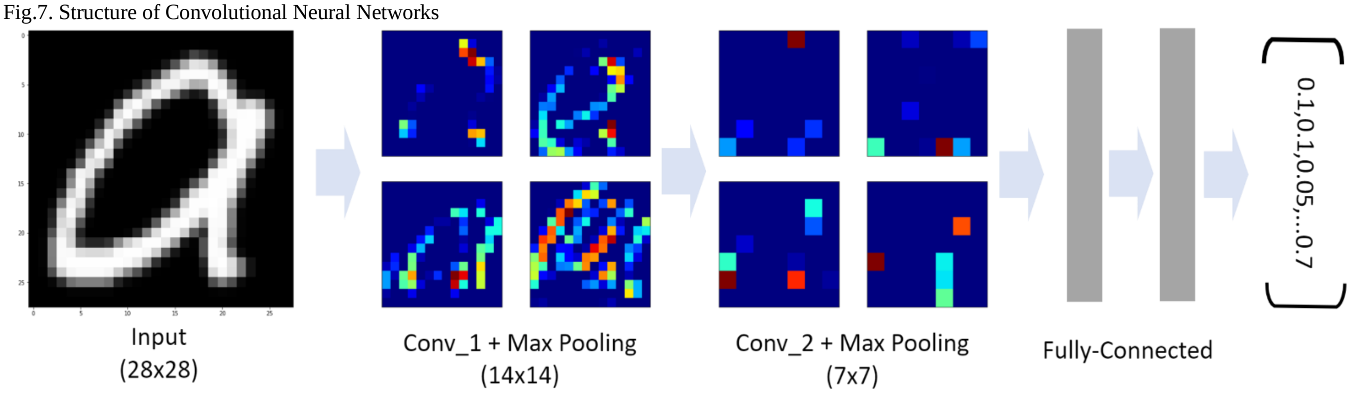


Fig.8. Visualization of feature maps in Convolutional Neural Networks(CNN)

data. The collected number of alphabet letter data is described as follows (A: 181, B:16, C: 222, D: 4, E: 64, F: 0, G:3, H:0, I: 175, J: 16, K: 4, L: 1414, M: 118, N: 172, O: 0, P: 4, Q: 0, R: 16, S:1, T: 2962, U: 0, V: 193, W:0, X:0, Y: 40, Z: 0).

V. CLASSIFICATION

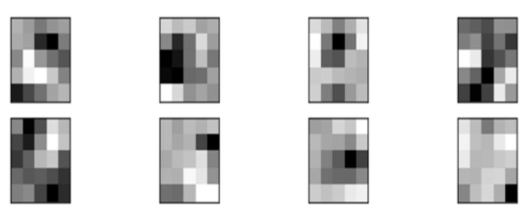


Fig.9. Visualization of filters in Conv_1

A. Network Structure: CNN with Keras

The configuration of our networks for digit and letter classification is specified. The network is constructed by taking the structure of a baseline network-CNN [12], two convolutional steps followed by one fully-connected layer for classification. Each convolutional step consists of one convolutional layer with padding to keep the size of input and one max-pooling layer to reduce the size of image and prevent overfitting. For the first convolutional layer, 5x5 filter was used to extract features of images. For the second convolutional step, 2x2 filter was applied and one dropout layer was also added to further prevent overfitting. When a 28x28 gray-scale digit or letter image enters into the network, the final feature map is going to shrink into 7x7 image with 64 channels through the CNN network. The CNN architecture is shown in Fig. 1 and all hyperparameters are described: In order to optimize the network, Adam optimizer [13] was applied and a learning rate of 0.01 was used. Data was trained for 50 epochs with a batch size of 128.

B. Data Preparation

Digit and letter images are all 28 x 28 gray-scale images and every pixel values ranging from 0 to 255. The dataset was divided by 255 for normalization, which speed up learning and leads to faster convergence. As described in section IV, oversampling was implemented to give a balance between minor classes and major classes. As the result, hand-written digit database and letter database become balanced datasets (digit: 10,000 train, 1,000 test / letter: 2,500 train, 125 test examples). After then, train sets for digits and letters were split into two parts respectively: a small fraction (digit: 2000 examples, letter: 500 examples) was used for the validation set to evaluate the model and the rest of data was allocated for training the model. For label encoding, all labels need to be converted to one-hot vectors. For example, 10-digit numbers have labels from 0 to 9 and label 2 represents one-hot vectors [0,0,1,0,0,0,0,0,0,0].

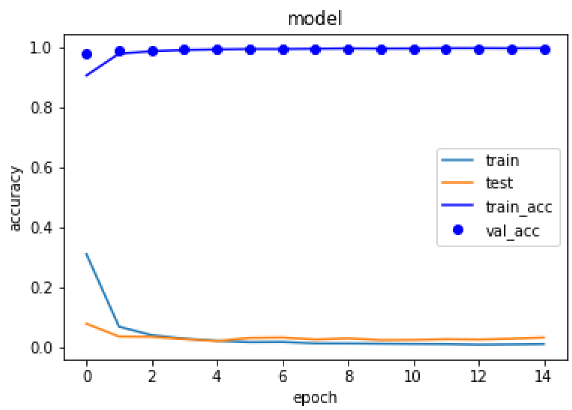
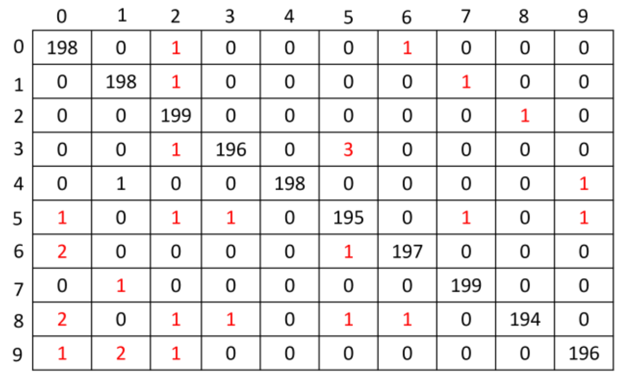


Fig.10. Accuracy and Loss Curves of CNN on train and test sets



	0	1	2	3	4	5	6	7	8	9
0	198	0	1	0	0	0	1	0	0	0
1	0	198	1	0	0	0	0	1	0	0
2	0	0	199	0	0	0	0	0	1	0
3	0	0	1	196	0	3	0	0	0	0
4	0	1	0	0	198	0	0	0	0	1
5	1	0	1	1	0	195	0	1	0	1
6	2	0	0	0	0	1	197	0	0	0
7	0	1	0	0	0	0	0	199	0	0
8	2	0	1	1	0	1	1	0	194	0
9	1	2	1	0	0	0	0	0	0	196

Fig.11. Confusion Matrix of Digit Test dataset

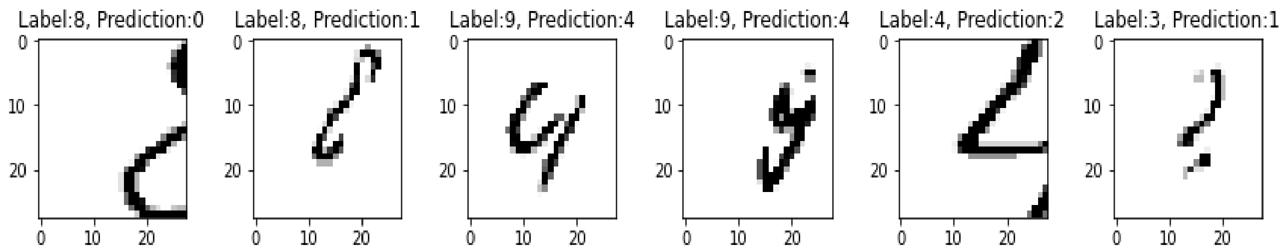


Fig.12. Misclassified digit images from test dataset.

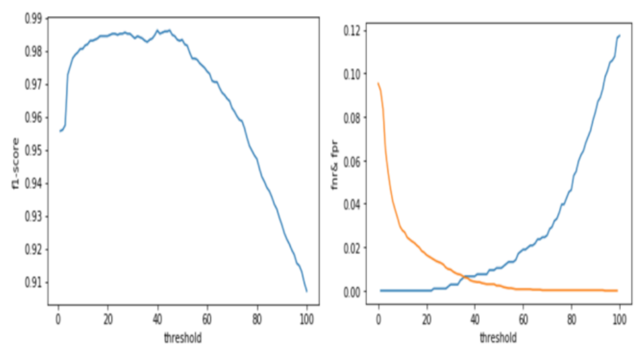


Fig.14. f1-score, True Positive Rate, and False Positive Rate by threshold

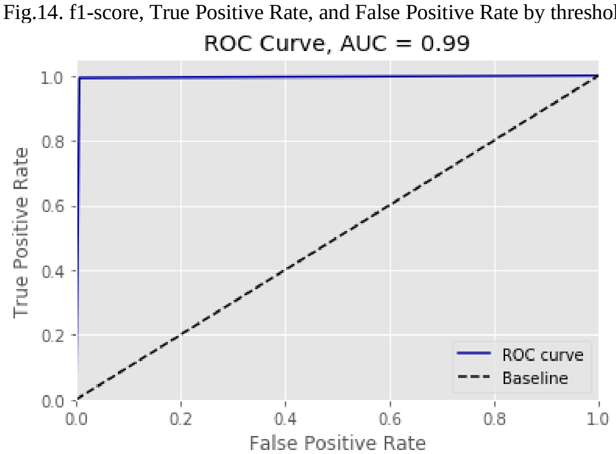


Fig.15. ROC curve at threshold = 36

C. Training

Training data is processed followed by the definition of networks mentioned in section V.(a). For learning, Keras 2.7.0 and Tensorflow 2.7.0 framework for deep learning and python 3.7.12 were utilized. GPU(Tesla K80, CUDA Version: 11.2) was used to speed up learning process. While training data points, early stopping was introduced to prevent overfitting and best parameters were saved as check point which allow to load the saved model after training.

D. Evaluation

Digit and letter classifiers were trained for 50 epochs respectively. For the digit classification, the training accuracy was 99.98 % and validation accuracy was 99.%. Test accuracy was 98.5% and you can check the misclassified digits from the confusion matrix. Figure 12 illustrates the wrong predicted test images. One-sided digit images usually cropped out of box has shown a lower digit recognition rate than the recognition rate of the centered images with margins. In the case of blurry writing style, character recognition rate was also lower than vivid and clear writing image.

-update: confusion matrix

VI. Result

A. Marked dot

After image processing described in section IV, checked mark or filled bubble would be a binary image without circled-boundary. To determine the mark is checked or not, the number of black pixels was counted and if the number exceeds a certain value - threshold, the mark will be identified as checked and recorded as yes (1). In order to distinguish the small dot or noise made by mistake in the check box and

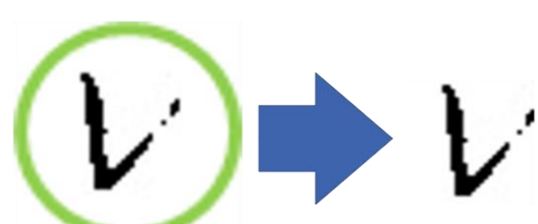


Fig.13. Image Processing for Check Box

bubbles, check marks, or 'X' marks, we had some experiment to set an optimal threshold by varying the cut-off point. To determine the point, f1-score, True positive rate(TPR, sensitivity), and False positive rate(FPR, specificity) were investigated and the graph of the result has shown in Fig.14. When the threshold is 36, sensitivity and specificity graphs meet at one point, and ROC curve at this point is very close to the top-left corner. In addition, AUC(Area under the roc curve) of 0.99 indicates the diagnostic ability of the marked dot classifier is outstanding.

B. Document Recognition

Total 125 scanned forms including TS01, TS02, TS04, and TS05 were used for evaluation of document reading. Every field in the form has a text box or checkbox to fill in. Depending on the patient's health status or response, nurse can leave the questions blank. As described in section III, all textboxes and check boxes should be detected through a series of processing techniques. In the next step, extracted letters, digits, or marked dot will be recognized by the pre-trained classifier. To distinguish the blank image, the number of black pixels was counted for both text and check box. In the whole pipeline, all recognized images are memorized as a pair of predicted label and position on a document, and then predicted labels(digit, letter, marked-dot) are sent to REDCap database.

As the result, 16,300 digits and 1,260 letters (including blank images) were detected and labeled. The accuracies for digit and letter were 95.73% and 95.08% respectively. Performance of popular NNs were also compared with our CNN model, and Table1 shows the result in accuracy. According to literature review, simple CNN based on PyTorch showed 98.89%, and more complicated neural network model provided more than 99.8% accuracy for EMNIST digit data [14], [15]. In our model, only collected digit images from Moldova were used for learning and evaluating the model. Test accuracy was 99.07% (in the experiment of combining EMNIST with our own data, test accuracy was 99.17%, but test accuracy using only Moldova data was much less than 90%) and 95.75% accuracy was achieved for new scanned files from Moldova. Even though the number presented by other NNs model is bigger than our number, the difference between the clean dataset(no noise, middle centered) and realistic data(arbitrarily located, a lot of noise, blurry) illustrates that out model has a good performance in digit recognition. For letter recognition, EMNIST letter dataset was utilized with letter images from Moldova together because of small number of letter data points. To test a performance in letter recognition, alphabet 'A', 'C', 'E', 'I', 'L', 'N','T', and 'V' were only used (there was no data point for the other letter characters in test document). Test accuracy was 98.4% and document-based accuracy using new scanned files was 95.08%.

Data	Model	Test Accuracy
EMNIST (Digits) [15]	CNN	98.89%
	CNN(Spinal FC)	99.12%
	VGG-5	99.81%
	VGG-5 (Spinal FC)	99.82%
Own Digit	Our Model	**99.07% (95.73%)[a]**
EMNIST (Letters) [15]	CNN	87.57%
	CNN(Spinal FC)	90.07%
	VGG-5	95.86%
	VGG-5 (Spinal FC)	95.88%
EMNIST + Own Letter	Our Model	**98.4% (95.08%)[a]**

Table.1. NNs' performance Comparison
[a] document-based test accuracy

VII. Conclusion

We presented text detection and recognition system for medical forms using Convolutional Neural Networks (CNNs). Given scanned medical forms on Tuberculosis, image processing techniques using OpenCV is applied to detect a set of bounding boxes containing digits and letters. Then character-level text boxes were extracted using the same processing techniques. These letter or digit images were inserted into the designed classifier (CNNs) for training. Evaluation in letter and digit recognition has presented good performance compared to the popular model, and to see the checkbox is checked or not, a simple binary classification is introduced by the number of black pixels. This binary classifier showed almost perfect performance (0.99 of AUC). The system can also reduce human-error in typing and redundant and repeated works by saving and retrieving digitized data on database. Therefore, the recognition system is expected to be a start toward digital health in nations that has limited resources and IT technology.

Based on this research, some points intriguing our interest were investigated. First, the collected number of letter images was so small to train because 'nurse initial' field has only letters on medical form and the very different writing style depending on culture and region degraded the maximum use of EMNIST. In order to obtain the more generalized letter classifier, more letter images need to be secured.

Second, we found that design of medical forms, data points and structure can make a significant effect on the performance in text recognition. In more details, in medical form, there is the inner vertical line between two characters in text boxes. The existence of vertical line bothers the text-level recognition.

Eliminating the vertical line between characters need more image processing techniques, and characters containing a long bar such as ‘1’, ‘7’, ‘I’, ‘J’, or ‘L’ can be affected and lose the features by applying the processing techniques. If there is no vertical line, new approach based on LSTM can be tested for the next research.

In addition, constructed dataset using the scanned files from Moldova has no bounding boxes information. If bounding box information can be given as ground truth, text detection also can be implemented by deep learning approach without manipulation of OpenCV. Our future goal is to obtain more data and improve the reliability of system performance. In addition, more diverse text detection and recognition system will be investigated.

Appendix

Appendixes, if needed, appear before the acknowledgment.

Acknowledgment

The preferred spelling of the word “acknowledgment” in American English is without an “e” after the “g.” Use the singular heading even if you have many acknowledgments. Avoid expressions such as “One of us (S.B.A.) would like to thank ... .” Instead, write “F. A. Author thanks ... .” **Sponsor and financial support acknowledgments are placed in the unnumbered footnote on the first page**.

References

[1] A. Singh, K. Bacchuwar, and A. Bhasin, “A survey of OCR applications,” *Int. J. Mach. Learn. Comput.*, vol. 2, no. 3, pp. 314–318, 2012.

[2] R. Smith, “An overview of the tesseract OCR engine,” in *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007) Vol 2*, 2007.

[3] *7Building a Multi-Lingual OCR Engine.pdf at main · tesseract-ocr/docs* .

[4] P. Gayathri and S. Ayyappan, “Off-line handwritten character recognition using Hidden Markov Model,” in *2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 2014.

[5] M. Rajnoha, R. Burget, and M. K. Dutta, “Offline handwritten text recognition using support vector machines,” in *2017 4th International Conference on Signal Processing and Integrated Networks (SPIN)*, 2017.

[6] F. Borisyuk, A. Gordo, and V. Sivakumar, “Rosetta: Large scale system for text detection and recognition in images,” in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018.

[7] P. G. Biondich, J. M. Overhage, P. R. Dexter, S. M. Downs, L. Lemmon, and C. J. McDonald, “A modern optical character recognition system in a real world clinical setting: some accuracy and feasibility observations,” *Proc. AMIA Symp.*, pp. 56–60, 2002.

[8] L. V. Rasmussen, P. L. Peissig, C. A. McCarty, and J. Starren, “Development of an optical character recognition pipeline for handwritten form fields from an electronic health record,” *J. Am. Med. Inform. Assoc.*, vol. 19, no. e1, pp. e90-5, 2012.

[9] P. L. Peissig *et al.*, “Importance of multi-modal approaches to effectively identify cataract cases from electronic health records,” *J. Am. Med. Inform. Assoc.*, vol. 19, no. 2, pp. 225–234, 2012.

[10] A. Baldominos, Y. Saez, and P. Isasi, “A survey of handwritten character recognition with MNIST and EMNIST,” *Appl. Sci. (Basel)*, vol. 9, no. 15, p. 3169, 2019.

[11] G. Cohen, S. Afshar, J. Tapson, and A. van Schaik, “EMNIST: an extension of MNIST to handwritten letters,” *arXiv [cs.CV]*, 2017.

[12] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” *Commun. ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[13] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv [cs.LG]*, 2014.

[14] H. M. D. Kabir *et al.*, “SpinalNet: Deep neural network with gradual input,” *arXiv [cs.CV]*, 2020.

[15] G. Cohen, S. Afshar, J. Tapson, and A. van Schaik, “EMNIST: Extending MNIST to handwritten letters,” in *2017 International Joint Conference on Neural Networks (IJCNN)*, 2017.

[16] M. Kumar and J. Mostafa, “Electronic health records for better health in the lower- and middle-income countries: A landscape study,” *Libr. hi tech*, vol. 38, no. 4, pp. 751–767, 2020.

[17] World Health Organization, “Global strategy on digial health 2020-2025,” 2021.

[18] “Review of Offline Handwriting Recognition Techniques in the fields of HCR and OCR,” *Int. J. Comput. Trends Techno*.

[19] M. M. Abu Ghosh and A. Y. Maghari, “A comparative study on handwriting digit recognition using neural networks,” in *2017 International Conference on Promising Electronic Technologies (ICPET)*, 2017.

[20] O. Morillot, L. Likforman-Sulem, and E. Grosicki, “Comparative study of HMM and BLSTM segmentation-free approaches for the recognition of handwritten text-lines,” in *2013 12th International Conference on Document Analysis and Recognition*, 2013.

[21] R. Li, H. Wang, and K. Ji, “Feature extraction and identification of handwritten characters,” in *2015 8th International Conference on Intelligent Networks and Intelligent Systems (ICINIS)*, 2015

[22] J. R. Prasad and U. V. Kulkarni, “Trends in Handwriting Recognition,” in *2010 3rd International Conference on Emerging Trends in Engineering and Technology*, 2010.

[23] M. Akram, R. Qasim, and M. A. Amin, “A comparative study of signature recognition problem using statistical features and artificial neural networks,” in *2012 International Conference on Informatics, Electronics & Vision (ICIEV)*, 2012.

[24] V. Romero et al., “The ESPOSALLES database: An ancient marriage license corpus for off-line handwriting recognition,” *Pattern Recognition.*, vol. 46, no. 6, pp. 1658–1669, 2013.

[25] Y. Chajri, A. Maarir, and B. Bouikhalene, “A comparative study of handwritten mathematical symbols recognition,” in *2016 13th International Conference on Computer Graphics, Imaging and Visualization (CGiV)*, 2016.

[26] U. Choudhary, S. Bhosale, S. Bhise, and P. Chilveri, “A survey: Cursive handwriting recognition techniques,” in *2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*, 2017.

[27] R. B. Arif, M. A. B. Siddique, M. M. R. Khan, and M. R. Oishe, “Study and observation of the variations of accuracies for handwritten digits recognition with various hidden layers and epochs using convolutional neural network,” in *2018 4th International Conference on Electrical Engineering and Information & Communication Technology (iCEEiCT)*, 2018.

[28] W. Xue, Q. Li, and Q. Xue, “Text detection and recognition for images of medical laboratory reports with a deep learning approach,” *IEEE Access*, vol. 8, pp. 407–416, 2020.

[29] S. Tabassum et al., “Recognition of doctors’ cursive handwritten medical words by using bidirectional LSTM and SRP data augmentation,” in *2021 IEEE Technology & Engineering Management Conference - Europe (TEMSCON-EUR)*, 2021.

[30] Centers for Disease Control and Prevention, “2019 National Electronic Health Records Survey Public Use File National Weighted Estimates,” 2019.

First A. Author (M’76–SM’81–F’87) and the other authors may include biographies at the end of regular papers. Biographies are often not included in conference-related papers. This author became a Member (M) of IEEE in 1976, a Senior Member (SM) in 1981, and a Fellow (F) in 1987. The first paragraph may contain a place and/or date of birth (list place, then date). Next, the author’s educational background is listed. The degrees should be listed with type of degree in what field, which institution, city, state or country, and year degree was earned. The author’s major field of study should be lower-cased.

The second paragraph uses the pronoun of the person (he or she) and not the author’s last name. It lists military and work experience, including summer and fellowship jobs. Job titles are capitalized. The current job must have a location; previous positions may be listed without one. Information concerning previous publications may be included. Try not to list more than three books or published articles. The format for listing publishers of a book within the biography is: title

of book (city, state: publisher name, year) similar to a reference. Current and previous research interests end the paragraph.

The third paragraph begins with the author's title and last name (e.g., Dr. Smith, Prof. Jones, Mr. Kajor, Ms. Hunter). List any memberships in professional societies other than the IEEE. Finally, list any awards and work for IEEE committees and publications. If a photograph is provided, the biography will be indented around it. The photograph is placed at the top left of the biography. Personal hobbies will be deleted from the biography.