Transfer learning based Optical Character Recognition using Natural Images

Shankar Lonare¹, Saket Jain²

¹,²Department of computer science & Engg , VNS Group of institution Bhopal

Abstract— Text detection and recognition has been a well-studied problem in the past. However, when it comes to detection and identification of text in natural images and text from natural scenes, it becomes a much more challenging problem because of the distortion in geometry, variance in the illumination. Recently, deep learning techniques achieved state-of-the-art performance in object detection and recognition. But deep learning required large data set and high computation power for training model from the scratch. Therefore, in this research work, a deep learning technique is used, which is based on the knowledge transfer of pre-trained Convolutional neural networks (CNNs) to recognize the text in the natural images.

In this work we have uses pre-trained VGG19 model. To fine-tuned the VGG19 model top layer is removed and new layer have been include in the network then it is trained (fine-tuned) on the optical character images. A CHAR74K dataset, which is a benchmark for natural images, is used for the evaluation of the proposed method. The proposed model has achieved an accuracy of 87.56 % and F1- score of 88%.

Keywords— Convolutional Neural Networks (CNN), Deep Learning, Optical Character Recognition (OCR), Maximally Stable Extremal Region (MSER), pre-trained network VGG19.

I. INTRODUCTION

Text plays an important role in vision-based applications. Text detection is basically the identification of text in a given set of images. In the most advanced computer vision applications such as visual assistance image retrieval, text reading in natural images is very important. It is because of text in images usually conveys important information. Text detection and identification has therefore received a great deal of attention in recent years. Over the past few years, Text detection and identification in natural images has evolved as an important and challenging problem. And it also has a lot of practical applications. Unlike recognition of characters in scanned documents, recognising text in natural and unconstrained images is a very complex problem which is complicated by fonts, conditions of lighting, textures and a lot of variations in backgrounds. Unfortunately, text characters in natural images can be of any gray-scale color (It does not always include white), variable length, low resolution and in complex backgrounds.

Natural picture text reading is important for many advanced applications such as image and video collection, scene understanding and visual support because image text typically conveys useful information. text reading is important. Therefore, in this culture, growing attention has been given to the identification and recognition of text in scene images. Although extensively studied in recent years, text detection in unconstrained environments is still quite difficult due to a number of factors, such as high variability in character font, width, colour, orientation, as well as complicated context and non-uniform lighting. Previous works in that area were used for text detection based on sliding windows and the study of the related components. Through rotating a multi-scaled grading panel, sliding window based methods identify text areas. Although this exhaustive search achieves high
recall levels, it is computer-inefficient. Methods focused on connected components derive characteristics through the analysis and the technique for grouping and refining of connected components. False alarm can therefore be removed and non-text elements eliminated. Stroke width transform (SWT) [1] and MSER [2] are two representative methods that achieve state-of-the-art output on a CHAR74 K dataset with MSER-based methods [3]. The MSER algorithms, however, extract massive non-text repetitive parts which are restricted by the false deletion and refining rules. These techniques can not also detect noise or distorted characters in the background. More recently, many deep learning-based approaches were developed for text scene detection due to profound model functionality representations. These models calculate high-level deep features from image patches or proposals for text/non-text classification based on convolutional Neural Networks (CNNs). These methods are also limited by the discriminative capacity of the local proposals and the CNN classifiers. In this project we suggest a flexible solution that incorporates the advantages of MSER and CNN characteristics.

II. LITERATURE REVIEW

The identification of character is not a new problem but can be traced back to structures in advance of computer developments. The first OCR systems were not computers, but mechanical devices that could identify characters, but were highly slow and of low precision. 1951, M. M. Sheppard has created a GISMO reading and robot that can be regarded in modern OCR [4] as the earliest work. J. Liang et al. [5] states: 'There is no need for OCR systems that handle special symbols to promote further growth.' If too many symbols are stored in the database it will limit efficiencies. The difficulty of introducing such a function is to make the system a general solution. The greatest trend in the identification of symbols is to analyze documents scanning the mathematical symbols. Mathematical language, which can understand symbols, was sponsored by Tesseract. This is an intelligent way to use the OCR engine to capture both natural and mathematical symbols. For road surface mapping applications, OCR can be used to detect symbols [6]. The paper states that the protocol for identification of road signs is identical to the facial recognition process (a device that recognizes faces). Tesseract again served the researchers as the motor of selection. The experiment with road symbols gave results with 80% defined symbols, with 2.5% incorrectly labeled.

The study, found Tesseracts to be ideal with high resolution and sharp images, yielded good results despite the poor quality of the camera and the vehicle was moving during the trial. The more complex symbols with the same contour and different interior were the most difficult symbols to be recognized. A separate study was required in order to distinguish symbols with the same layout with different insides. Even signs of distinct outlines could be identified and analyzed. Tesseract was launched in HP Laboratories, Bristol as a PhD research project. It became popular and between 1984 and 1994 was produced by HP. Tesseract was released as open source software in 2005 [7]. Instead it uses the classifier to identify damaged characters, to replace the entire character. The engine is not trained to recognize damaged information. This could mean a much smaller and more efficient repository of the learning samples. Initially, it was equipped with 20 samples of 94 different characters, 8 different fonts with 4 characteristics, with a maximum sample of 60160 samples (normal, Bold, Italic, Bold Italic).

One engine, Tesseract, was mentioned repeatedly during the OCR engine examination. Unless the tests are correct [8], signs [6] are also used for analysis. The engine looks promising, but is insufficient to make specific conclusions. Tesseract [9] supports UTF 8, and can recognize over 100 languages from the container, with more languages still being supported. The motor is trainable, so it is possible to train and understand a new language or font which is normally not sponsored. Tesseract is known as the market's most precise open source motor. For the fundamental collection, Leptonica uses the photo care library [10]. The software is created and built in C or C++, but other
programs can be used in other languages. The Apache license 2.0 [11] is valid. A third-party wrapper: JavaCPP along with its various presents [12] is used to work with Tesseract in Java. Tan Chang Wei et al. [13], used deep neural network for the learning and execution of OCR using Inception V3. The V3 network Inception consists of 53,342 noisy pictures from receives and newspapers. Fabio De Sousa Ribeiro et al. [14] a dual deep, neural network-based system for automatically identifying dates of usage on the food package photos is proposed for an end-to-end architecture. The system includes: a global neural convolution (CNN) network to assess the value of food packets (blurry / clear / missing use by date data); and a completely convolutional network at the local level (FCN) for ROI position to use by date.

Choudhary Savita et al. [15] Proposes a text area detection method with MSRs and a self-trained neural network to recognize the message. MSR theory. MSR theory. The photo is prefabricated, MSER and the canny edge are used to find the smaller areas that can more likely contain text. The text is extracted as a single character by simple algorithms in the binary image and the method for recognition is passed through which dim characters are specifically designed.

OCR systems lie in document recognition where it is impossible to control the production process. This can occur where the receiver is removed from an electronic version and does not monitor the production process or older content which cannot be digitally processed at production time. In order to read the impressed text, future OCR-systems must be omnivorous. The identification of hand-produced documents is another important field for OCR. OCR will concentrate, for example, on the reading of addresses in email from persons without access to computer software in the context of postal applications. It’s no longer rare for companies, etc. to mark mails with barcodes, with access to computer technology. The relative importance of handwritten text recognition is therefore expected to increase.

III. METHODOLOGY

In this chapter, we are presenting detail description of the proposed method for optical character recognition (OCR).

Figure 1: Layerd architecture of VGG19.
3.1 Proposed Method
In this section, we describe the proposed deep learning method. Deep learning techniques are a very popular and powerful method to solve the pattern recognition problem in computer vision. The proposed method is based on the transfer learning-based approach in which pre-trained VGG19 convolutional neural network is fine-tuned to predict the OCR. Detail descriptions proposed approach is given in the following sections. VGG19 is very popular pre-trained network, which trained on the million of images of Imagenet dataset. VGG19 is able to categorizes the 1000 categories of objects. Its architecture is shown in the figure 1. In my situation, I chose the VGG19 model for some reasons. First, it took the 2nd place showing nice performance. I only need 62 categories of images, so I though VGG19 is enough for OCR. Second, VGG19 architecture is very basic. If you understand the basic CNN model, you can immediately find that VGG19 looks similar.

3.2 Algorithm
Steps that are used in our proposed methodology is describes in this section. In the proposed transfer learning based model is fine-tuned on the optical characters. Once model is trained then it is able to recognize the characters. Main steps of proposed method are as follows:

| **Input:** | Query and database images |
| **Output:** | Retrieved Images |

**Step 1:** Read all images from database.

**Step 2:** Load pre-trained VGG19 model.

**Step 3:** Remove top layers (such as Fully connected, Softmax and Output layer) from the VGG19 model.

**Step 4:** Add new Fully connected with 62 neuron, Softmax and Output layer.

**Step 5:** Set learning false (by using rate 0.00001) of base model and top layers learning rate to 0.001.

**Step 6:** Now fine-tuned network by using Adam optimizer on the CHAR74k dataset of optical characters.

**Step 7:** Apply test images on the fine-tuned model and recognize the characters.

**Step 8:** Now evaluate the performance of the model.

IV. IMPLEMENTATION AND RESULT ANALYSIS
This Section contains the codes of this project. We implemented problem statement using python 3.6. The editor used was Anaconda. For visualization and graphs we have used standard matplotlib Library of python. For implementing Neural Networks we have used Keras Library with Tensorflow2.0 as the backend.

4.1 Dataset
Character recognition is a classic pattern recognition problem that scientists have been working on since computer vision's early days. With today's cameras omnipresence, automatic character recognition applications are wider than ever before.
This is largely considered a solved problem in restricted circumstances of Latin script, such as images of scanned documents with common character fonts and uniform context. Photos captured with common cameras and hand held phones, however, still present a formidable challenge for the identification of characters. In this data set, the challenging aspects of this issue are evident. Char74K consist 7,712 optical character images of 62 classes. In which 10 classes of numbers from 0 to 9 and 52 classes of 26 lower case and 26 upper case characters. Sample image of Char74K dataset is shown in the figure 2. We have divided Char74K dataset randomly into a training set, validation set and test set in the ratio of 7:20:10.

| Set         | Total no of images |
|-------------|--------------------|
| Training Set | 5403               |
| Test set    | 1543               |
| Validation  | 766                |

Table 1: Number images in the training, test and validation set.

Training set is used to train the VGG19 model, validation set is used for selecting based parameters of the network and test set is used for performance analysis of the proposed model.

4.2 Implementation and experimental setup

For the implementation of the proposed method, we have used Tensorflow 2.0 and Keras library. Tensorflow is very powerful research tool which provides many built-in images processing toolbox and deep learning. Hyper parameters are initialized as learning rate to 0.01, mini-batch size to 32. We found these hyper parameter values experimentally that are well suited to our issue. Network is train for 200 epochs. Figure 4.3 shows the proposed network training performances.

4.3 Result

In this section, we present the results obtained from the experiment over the database of Char74 dataset. Table 2 shows the performances of the model on the training, validation and test set. As observed from the table network performance on the validation set and test are the approximately same, therefore we can say our model is stable.
Table 2: Network performance of the Char74K dataset.

| Set           | Accuracy (in %) |
|---------------|-----------------|
| Training Set  | 93.24           |
| Test set      | 87.56           |
| Validation    | 88.65           |

Table 3: Performance of proposed approach in terms of precision recall and F1-Score on individual classes.

| Class         | Precision | Recall | F1-score | No. of sample |
|---------------|-----------|--------|----------|---------------|
| Digit         | 0.99      | 0.84   | 0.9      | 119           |
| Capital Letter| 0.94      | 0.9    | 0.92     | 1042          |
| Small Letter  | 0.92      | 0.83   | 0.87     | 382           |
| All Classes   | 0.89      | 0.88   | 0.88     | 1543          |

To evaluate the performance of the proposed method, we have tested three subset of test set as in first present only digit images (for 10 categories); in second Capital letter images (for 26 categories) and in the third subset we present small letters images (for 26 categories) and overall performance of the model on the test set (for all 62 categories) in terms of precision, recall and F1-score is shown in the Table 3.

V. CONCLUSION

In this paper, we have used the architecture and weights of pre-trained VGG19 model. We have first fine-tuned the proposed model on the optical character Char74K dataset. To fine-tuned the VGG19 model top layer is removed and new layer have been include in the network then it is trained (fine-tuned) on the optical character images. Our proposed system exhibits good performance over CHAR74K character dataset and it has achieved an accuracy of 87.56 % and F1-score of 88%.

REFERENCES

I. Epshtein, Boris, EyalOfek, and Yonatan Wexler. "Detecting text in natural scenes with stroke width transform." In 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 2963-2970. IEEE, 2010.

II. Gou, Chao, Kunfeng Wang, Zhongdong Yu, and HaitaoXie. "License plate recognition using MSER and HOG based on ELM." In Proceedings of 2014 IEEE International Conference on Service Operations and Logistics, and Informatics, pp. 217-221. IEEE, 2014.

III. De Campos, T., R. B. Bodla, and M. Varma. "The chars74k dataset." (2009).

IV. D. C. Ciresan, U. Meier, J. Masci, L. M. Gambardella, and J. Schmidhuber. High performance neural networks for visual object classification. Technical Report IDSIA-01-11, DalleMolle Institute for Artificial Intelligence, 2011.

V. Jisheng Liang, Ihsin T Phillips, Vikram Chalana, and Robert Haralick. A methodology for special symbol recognitions. In Pattern Recognition, 2000. Proceedings. 15thInternational Conference on, volume 4, pages 11-14. IEEE, 2000.

VI. Markus Schreiber, Fabian Poggenhans, and Christoph Stiller. Detecting symbols onroad surface for mapping and localization using ocr. In Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on, pages 597-602. IEEE, 2014.

VII. Ray Smith. An overview of the tesseract ocr engine. In icdar, pages 629-633. IEEE, 2007.

VIII. Ocr software comparison tesseract vs abbyyfinereader. 2015. http://lib.psnc.pl/dlibra/docmetadata?id=358&from=publication&showContent=true. Accessed: 2016-02-27.

IX. Tesseract optical character recognition engine. https://github.com/tesseract-ocr. Accessed: 2016-02-05.

X. Leptonica, useful for image processing and image analysis applications. http://www.leptonica.com/ Accessed: 2016-04-06.

XI. Apache license 2.0. http://www.apache.org/licenses/LICENSE-2.0. Accessed: 2016-02-26.

XII. Javacpp- presets module contains java configuration and interface classes for widelyused c/c++ libraries. https://github.com/bytedeco/javacpp-presets.
XIII. Accessed: 2016-02-26.
XIV. Wei, Tan Chiang, U. U. Sheikh, and Ab Al-Hadi Ab Rahman. "Improved optical character recognition with deep neural network." In 2018 IEEE 14th International Colloquium on Signal Processing & Its Applications (CSPA), pp. 245-249. IEEE, 2018.
XV. Ribeiro, Fabio De Sousa, Liyun Gong, Francesco Calivá, Mark Swainson, KjartanGudmundsson, Miao Yu, Georgios Leontidis, Xujiong Ye, and Stefanos Kollias. "An end-to-end deep neural architecture for optical character verification and recognition in retail food packaging." In 2018 25th IEEE International Conference on Image Processing (ICIP), pp. 2376-2380. IEEE, 2018.
XVI. Choudhary, Savita, Nikhil Kumar Singh, and Sanjay Chichadwani. "Text Detection and Recognition from Scene Images using MSER and CNN." In 2018 Second International Conference on Advances in Electronics, Computers and Communications (ICAEC), pp. 1-4. IEEE, 2018.