An algorithm to identify text marked in rock core pictures with machine learning algorithm

Yunfeng Bai
Severnaya Dvina Emb. 17, Arkhangelsk, Russia
E-mail: yumengleme@126.com

Berezovsky, Vladimir
Severnaya Dvina Emb. 17, Arkhangelsk, Russia
E-mail: v.berezovsky@narfu.ru

Abstract. The aim of our project is to extract numerical descriptors or gather feature of rock samples from their images made under microscope with computer vision algorithm which can overcome the bottleneck of traditional method. It is impossible to solve it without getting the information of microscope magnification of images when they were made. That is why this paper is about an algorithm to identify the text used to mark microscope magnification information in rock core sample pictures. This is a sub project. As locations of text in these pictures are not fixed, font of text is not fixed, font size of text is not fixed, a method of two steps is designed to identify the text from the background. It works as follows: first, find the location of text; second, recognize the text with Convolutional Neural Network. Experimental results demonstrate the potential value of the proposed approach to identify the magnification information in rock core sample pictures and to apply as a function to a program of our future project.

1. Introduction
The aim of our project is to extract numerical descriptors or gather feature of rock samples from their images made under microscope with computer vision algorithm, which can efficiently save time compared with traditional method. The first question we met in this project is to identify the text which used to mark microscope magnification information in rock core sample pictures. As it is impossible to extract numerical descriptors or gather feature of rock samples from their images without getting the information of microscope magnification of these images when they were made. A method of two steps is designed to identify the text from the background. It works as follows: first, find the location of text; second, recognize the text with Convolutional Neural Networks, which is a most popular machine learning algorithm these days.

The paper is organized as follows. Section 2 gives an overview of related works in the field of identification text in images. Section 3 makes an review of Convolutional Neural Networks. Section 4 describes Tensorflow which is a open source machine learning framework made by Google. Section 5 presents the individual steps of our approach to text localization and text recognition. Section 6 contains the experimental results obtained and discussion. Section 7 concludes the paper and outlines areas for future research.
2. Related works

In 2003, an approach was proposed by Julinda Gllavata, Ralph Ewerth and Bernd Freisleben to detect, localize and extract texts from color images with complex backgrounds [1]. It works as follows: Color images are first converted to grayscale images [1]. An edge image is generated using a contrast segmentation algorithm, which in turn uses the contrast of the character contour pixels to their neighboring pixels [1]. This is followed by the analysis of the horizontal projection of the edge image in order to localize the possible text areas [1]. After applying several heuristics to enhance the resulting image created in the previous step, an output image is generated that shows the text appearing in the input image with a simplified background [1]. These images are ready to be passed to OCR (Optical Character Recognition) system. Experimental results on a set of images demonstrated the performance of their approach [1].

In 2009, an approach was proposed by Weijuan Wen, Xianglin Huang, Lifang Yang, Zhao Yang and Pengju Zhang for text location and segmentation [2]. It works as follows: Firstly, the image is converted into wavelet transform domain [2]. Then three sub-band wavelet images of HxLy, LxHy, HxHy are binarized and merged into a texture image by linear combination [2]. The texture image is processed by CRLA (constrained run length algorithm) and image smoothing to enhance the candidate text regions [2]. Finally, the text regions are located after 8-connected component growth and filtering out non-text regions [2]. Then the characters are segmented from text blocks based on the mean values of some slices of text-block are recognized by OCR [2]. Experimental results show that their method is robust to overlapped complex background [2].

In 2013, an algorithm was proposed by Anubhav Kumar and Neeta Awasthi for text localization and extraction for detection of both graphics and scene text in video images [3]. It works in four steps: Edge generation using Line edge detection mask, Text localization using projection profiles based method, Text segmentation and Text recognition [3]. The proposed technique gives better result than existing methods in terms of detection rate for large video image database with very few false alarms, reliable recall rate and precision rate [3].

In 2014, a connected component based approach was proposed by Kamrul Hasan Talukder and Tania Mallick to identify texts in images [4]. The text extraction process starts with conversion of the color image to gray scale image and then it converts the gray scale image into a binary image [4]. Then each text region is marked and the text is extracted from the image [4]. Finally, the extracted text is written into another gray scale image [4]. The experimental results demonstrate that the performance of the proposed method is valuable [4].

In 2017, Zhuoyao Zhong, Lei Sun and Qiang Huo proposed a LocNet to improve the localization accuracy of a Faster R-CNN based text detector [5]. Given a proposal generated by region proposal network (RPN), instead of predicting directly the bounding box coordinates of the concerned text instance, the proposal is enlarged to create a search region so that conditional probabilities to each row and column of this search region can be assigned, which are then used to infer accurately the concerned bounding box [5]. Experiments demonstrate that approach boosted the localization accuracy for Faster R-CNN based text detection significantly [5].

In 2015, an algorithm was proposed by T Kumuda and L Basavaraj for detecting and localizing text in natural scene images [6]. The entire work is divided into two stages. Text regions are detected in the first stage using texture features [6]. Discriminative functions are used to filter out non-text regions [6]. In the second stage the detected text regions are merged and localized [6]. An experimental results obtained showed that approach worked efficiently [6].

In 2014, an end-to-end text recognition system was presented by Michael O, Markus D, Stefan F, Florian K and Robert S [7]. The text recognition is done using a deep Convolution Neural Network (CNN) trained with backpropagation [7]. The system presented outperforms state of the art methods on the ICDAR 2003 dataset in the text-detection [7].

In 2016, Xinhao Liu, Takahito Kawanishi, Xiaomeng Wu and Kunio Kashino proposed a
highly accurate character recognition model by utilizing the representational power of a specially
designed Convolutional Neural Network (CNN) [8]. Based on the recognition model, they also
developed an efficient post processing approach for error correction and hypothesis re-verification
[8]. Character and word image recognition experiments on two public datasets show that the
proposed approach provided superior or comparable results to the state-of-the-art techniques
[8].

In 2018, a Rotational Region CNN was proposed by Yingying Jiang, Xiangyu Zhu, Xiaobing
Wang, Shuli Yang, Wei Li and Hua Wang to estimate approximate text regions [9]. Their
work follows three steps. First, a novel multi-task regression method was applied to support
arbitrarily-oriented scene text detection [9]. Second, multiple ROIPoolings was introduced to
address the scene text detection problem for the first time [9]. Third, an inclined Non-Maximum
Suppression (NMS) was applied to post-process the detection candidates [9]. Experiments show
that their method outperforms the state-of-the-art on standard benchmarks: ICDAR 2013,
ICDAR 2015, COCO-Text and MSRA-TD500 [9].

3. Convolutional Neural Networks (CNN)
The CNN was first proposed by LeCun [10]. It simulates the processing system of human
vision by using the local receptive field, shared weight, and subsampling [11]. Nowadays CNN
is one of most popular artificial neural networks and widely used in computer vision, natural
language processing, and speech recognition. A CNN is a multi-layered non-fully-connected
neural network [11] composed by the input layers, the hidden layers and output layers. The
hidden layer is of many neurons and connections between the input and output layers. The
hidden layer can be composed of several convolutional layers, several pooling layers, and several
full-connection layers in practice.

3.1. Convolutional layer
A convolutional layer is parametrized by the size and the number of the maps, kernel sizes,
skipping factors, and the connection table [12]. Each layer has M maps of equal size (Mx, My).
A kernel of size (Kx, Ky) is shifted over the valid region of the input image. The skipping
factors Sx and Sy define how many pixels the filter/kernel skips in x and y direction between
subsequent convolutions. The size of the output map is then defined as:

\[ M^n_x = \frac{M^{n-1}_x - K^n_x}{S^n_x} + 1 \]  

\[ M^n_y = \frac{M^{n-1}_y - K^n_y}{S^n_y} + 1 \]  

Where index n indicates the layer. Each map in layer Ln is connected to at most Mn-1 maps
in layer Ln-1. Neurons of a given map share their weights but have different receptive fields.

3.2. Pooling layers
The purpose of the pooling layers is to achieve spatial invariance by reducing the resolution of
the feature maps [13]. Each pooled feature map corresponds to one feature map of the previous
layer [13]. There are two kinds of pooling layers: max pooling and subsampling.

The max pooling function

\[ a_j = \max_{N_x \times N_y} (a^n_i u(n, n)) \]  

applies a window function u(x,y) to the input patch, and computes the maximum in the
neighborhood [13].
The subsampling function takes the average over the inputs, multiplies it with a trainable scalar $\beta$, adds a trainable bias $b$, and passes the result through the non-linearity \[13\].

In 2010, it has been found by Dominik Scherer, Adreas Muller and Sven Behnke that max-pooling can lead to faster convergence, select superior invariant features, and improve generalization \[13\].

### 3.3. Fully Connected Layers

The fully connected layer, as one of the important components of neural network, is mainly composed of two parts: linear operation and nonlinear operation. The linear transformation is a linear transformation from the perspective of the operation process. For an input vector $\vec{x} = [x_1, x_2, \ldots, x_n]^T$, it is transformed into $\vec{z} = [z_1, z_2, \ldots, z_n]^T$ by the matrix $W$, sometimes with an offset term $\vec{b} = [b_1, b_2, \ldots, b_n]^T$.

$$W\vec{x} + \vec{b} = \vec{z} \quad (4)$$

The linear part passes the summarized result to the nonlinear part, and the nonlinear part normalizes the obtained data. There will be problems when calculating from forward and reverse without the operation. Another important role of the nonlinear part is to break the previous linear mapping relationship. Suppose there is a two-layer fully connected neural network with no nonlinear layers, then for the first layer:

$$W^0 \vec{x}^0 + \vec{b}^0 = \vec{z}^1 \quad (5)$$

For the second layer:

$$W^1 \vec{x}^1 + \vec{b}^1 = \vec{z}^2 \quad (6)$$

Then:

$$W^1 W^0 \vec{x}^0 + \vec{b}^1 + \vec{b}^1 = \vec{z}^3 \quad (7)$$

$$W^1 W^2 \vec{x}^0 + W^1 \vec{b}^0 + \vec{b}^1 = \vec{z}^2 \quad (8)$$

It is clear that the role of multi-layer network is no different from the function of a layer of network if it is without nonlinear part \[14\].

### 4. TensorFlow

TensorFlow is Google’s second-generation machine learning algorithm implementation framework and Google chose to open up TensorFlow on GitHub in 2015 and released TensorFlow 1.0 in January 2017 \[15\]. It is a machine learning system that operates at large scale and in heterogeneous environments \[16\]. The front end of TensorFlow supports Python, C++, Go, Java and other development languages, and the back end is written in C++, CUDA, etc.

### 5. Proposed methods

Motivated by methods \[1-9\], an method is proposed to indentify text which indicated microscope magnification in the picture of rock core samples. As there are 11 kinds of characters need to be recognized and the code of this method will be one funtion of a program which is used to extract numerical descriptors or gather feature of rock samples from their images in future, this code should be as small as possible. Then a method was proposed. It includes two steps: first, find the location of text; second, recognize the text with Convolutional Neural Networks. For the first step, text regions are detected based on texture features. For the second step, CNN is composed of 5 layers which are include 2 convolutional layers, 2 pooling layers, 1 full-connection layers.
6. Experiment result and discussion

In our experiments, TensorFlow is used as a framework to implement CNN mentioned above. Weights parameters were initialized before training networks with function of TensorFlow. Least squares method was considered as the cost function to minimize in this experiment. Gradient decent algorithm was used to calculate the gradient of the cost function and update the weights parameters by iterating.

The networks designed were training with general-purpose GPU. The GPU used in the experiment is GeForce GTX 1050 Ti. It has 768 NVIDIA CUDA Cores and 4G GDDR5 memory which is of 7 Gbps Speed. The GPU Architecture is Pascal. The method was applied to rock image dataset which includes about 400 pictures. The correct rate is about 70%. It should be higher than this after optimization.

7. Conclusion

Experimental results demonstrate potential value of the proposed approach to identify the magnification information in rock core sample pictures. And it is small enough to apply as a function to program of our future project whose goal is to extract numerical descriptors or gather feature of rock samples from their images.

8. Acknowledgments

This work is supported by the Russian Foundation for Basic Research (RFBR) project No.16-29-15116 and China Scholarship Council.

References

[1] Gllavata J, Ewerth R and Freisleben B 2003 A robust algorithm for text detection in images Proc. Int. Conf. on 3rd International Symposium on Image and Signal Processing and Analysis 2003 (IEEE)
[2] Wei Juan W, Xiang Lin H, LIFANG Y, ZHAO Y and PENGJU Z 2009 An Efficient Method for Text Location and Segmentation Proc. Int. Conf. on 2009 WRI World Congress on Software Engineering (IEEE)
[3] Anubhav K and Neeta A 2013 An efficient algorithm for text localization and extraction in complex text images Proc. Int. Conf. on 2013 2nd International Conference on Information Management in the Knowledge Economy (IEEE)
[4] Kamrul T and Tanjia M 2014 Connected component based approach for text extraction from color image Proc. Int. Conf. on 2014 17th International Conference on Computer and Information Technology (ICCCIT) (IEEE)
[5] Zhuyao Z, Lei S and Qiang H 2017 Improved Localization Accuracy by LocNet for Faster R-CNN Based Text Detection Proc. Int. Conf. on 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR) (IEEE)
[6] Kumuda T and Basavaraj L 2015 Detection and localization of text from natural scene images using texture features Proc. Int. Conf. on 2015 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC) (IEEE)
[7] Michael O, Markus D, Stefan F, Florian K and Robert S 2014 End-to-End Text Recognition Using Local Ternary Patterns, MSER and Deep Convolutional Nets Proc. Int. Conf. on 2014 11th IAPR International Workshop on Document Analysis Systems (IEEE)
[8] Xinhao L, Takahito K, Xiaomeng W and Kunio K 2016 Scene text recognition with high performance CNN classifier and efficient word inference Proc. Int. Conf. on 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (IEEE)
[9] Yingying J, Xiangyu Z, Xiaobing W, Shuli Yang, Wei L and Hua W 2009 R2CNN: Rotational Region CNN for Arbitrarily-Oriented Scene Text Detection Proc. Int. Conf. on 2009 WRI World Congress on Software Engineering (IEEE)
[10] Lcun Y, Bottou L, Bengio Y and Haffner P 1998 Gradient-based learning applied to document recognition (IEEE) 2278-324
[11] Min P, Chongyang W, Tong C and Guangyuan L 2016 NIRFaceNet: A Convolutional Neural Network for Near-Infrared Face Identification 7
[12] Dan C, Ueli M, Jonathan M and Luca M 2011 Flexible, High Performance Convolutional Neural Networks for Image Classification Proc. Int. Conf. on Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence 2 1237-42
[13] Dominik S, Adreas M and Sven B 2010 Evaluation of pooling operations in convolutional architectures for object recognition Proc. Int. Conf. on International Conference on Artificial Neural Networks (Springer) 92-101

[14] Chao F 2017 Deep learning in a easy way——learn core algorithms and visual practice (Beijing: Publishing House of Electronics Industry) 50

[15] Wenjian H and Yuan T 2017 TensorFlow combat (Beijing: Publishing House of Electronics Industry) 1-2

[16] Abadi M, Paul B, Jianmin Chen and Zhifeng C 2009 TensorFlow: A System for Large-Scale Machine Learning TensorFlow: A system for large-scale machine learning Proc. Int. Conf. on 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI16) 265-84