2016 15th IEEE International Conference on Machine Learning and Applications

Recognition of Slab Identification Numbers using a Deep Convolutional Neural Network

Sang Jun Lee
Department of Electrical Engineering
POSTECH
Pohang, 790-784, Korea
Email: lsj4u0208@postech.ac.kr

Sang Woo Kim
Department of Electrical Engineering,
Department of Creative IT Excellence Engineering
and Future IT Innovation Laboratory
POSTECH
Pohang, 790-784, Korea
Email: swkim@postech.ac.kr

***Abstract*—In the steel industries, automated identification of product information is important for an efficient manufacturing process. This paper focuses on the recognition problem for slab identification numbers in factory scenes. The recognition problem in an actual industrial setting is significantly more challenging than character recognition in documents or natural scenes. The objective of this paper is to develop an end-to-end recognition algorithm for slab identification numbers, and a Deep Convolutional Neural Network (DCNN) was utilized to construct an integrated recognition algorithm. The proposed algorithm contains composition of training data and a DCNN model, and a decoding process is proposed to transcribe slab identification numbers. The proposed deep learning based algorithm showed a reliable recognition performance for actual industrial scenes.**

I. Introduction

A slab is a major semi-finished product in the steel industry. Slabs are produced by continuous casting in a steel factory, and further hot and cold rolling processes are conducted. In a manufacturing process, slab identification is necessary for efficient production logistics and stock control, and an automatic identification system enhances system reliability with reduced manual labor. Several auto-identification methods such as bar-code tagging and radio-frequency identification systems are used to identify product information in industrial fields, and paint marking machines with specialized paint that is endurable to the high temperature up to 1400°C are widely used to inscribe slab identification numbers because of the high temperature of slabs. Computer vision systems are utilized to identify slab information. Several issues are arisen for effective paint marking machines, image acquisition systems, and image processing algorithms. This paper focuses on a recognition algorithm.

In our actual industrial setting, a slab identification number consists of 9 characters: the first character is an alphabet that represents a type of a manufacturing line, and successive 8 characters are decimal numbers. Factory scenes were collected for a processing line, and initial characters of slab identification numbers are *B*.

The recognition of slab identification numbers in an actual factory scene is a difficult problem compared to recognitions in general unstructured images. In our industrial setting, various number of slabs with different sizes are piled up on a slab transfer machine. A product identification number can be blurred by a shadow casted by the upper slab or reddish color appeared on a hot slab, and a part of characters are deformed or removed by the high temperature of slabs. Edges of characters in an identification number are not obvious unlike general texts in natural scenes. Further difficulties are arisen by the change of background conditions and their complexities. Background contents of factory scenes are varied as steel production is being processed, and the lighting condition is changed in day and night. Huge amount of dust particles in the steel factory is another obstacle to obtain a clear factory scene. The objective of this paper is to develop a recognition algorithm for slab identification numbers in actual factory scenes.

Many papers were proposed to develop an algorithm to separately detect and recognize texts in a scene [1]–[6]. Although this step-wise approach is computationally efficient especially for characters with various sizes and orientations, errors in each step are accumulated [7]. In this paper, an integrated methodology, which conducts the localization and recognition with a combined module, is utilized to reduce the accumulated error. The most relevant paper to our research was proposed by Choi [8], and this paper proposed a rule-based algorithm for the localization and segmentation of slab identification numbers in actual factory scenes. We propose an integrated end-to-end algorithm for recognizing slab identification numbers with the use of a deep convolutional neural network. Actual factory scenes, which were used in the previous work [8], was utilized to develop and evaluate our proposed algorithm.

The remaining sections are organized as follows: Section II explains the proposed algorithm, and Section III shows an experimental result. Current limitations of the proposed algorithm are discussed in Section IV, and Section V contains conclusions.

II. Proposed Algorithm

The proposed algorithm contains a procedure for constructing training data and a model of Deep Convolutional Neural Network (DCNN). In the test phase, an input factory scene is resized for considering various sizes of identification numbers, and patches are loosely collected and classified to estimate

978-1-5090-6167-9/16 $31.00 © 2016 IEEE
DOI 10.1109/ICMLA.2016.146

vertical positions and widths of identification numbers. A sub-image that has the closest width to a desired value is extracted from resized images for an identification number, and patches from the sub-image are densely collected and classified to obtain a character response map. A character response map is decoded to transcribe a slab identification number, and detailed explanations are described in the following subsections.

A. Training Data

From an actual steel manufacturing process, factory scenes were recorded as color images with the size of 1200 (height) × 1920 (width) × 3 (channel), and 23571 characters from 2619 slab identification numbers in 1295 images were collected to generate training data. Data augmentation was conducted for the characters to obtain sufficient number of training data and to achieve robust generalization performance. For each character, 5% and 10% of top, bottom, left, right, top-left, top-right, left-bottom, right-bottom regions were cropped and expanded, and totally 78 million character images were collected. The average height and width of the original characters were 80.18 and 55.98 pixels, and the standard deviations of heights and widths were 16.61 and 10.76 pixels. The original and augmented character images were resized to 80 × 56. Background training patches were collected from the 1295 factory scenes by using a non-overlapping sliding window method with the size of 80 × 56. If the center point of a background patch belongs to a character region, this patch is extracted from the set of background patches, and 52 million background patches were obtained.

B. Deep Convolutional Neural Network

This section presents an architecture of a DCNN, and it is shown in Fig. 1. The input of the DCNN is a character or background patch image with the size of 80 (height) × 56 (width) × 3 (channel), and the output of the soft-max layer is a 12-dimensional probabilistic vector. The $1^{st} \sim 10^{th}$, 11^{th}, and 12^{th} components of an output vector represent the probabilities that the corresponding patch belongs to a numerical character in *0* $\sim$ *9*, the alphabetic character *B*, and a background patch, respectively.

In Fig. 1, a box represented by $conv\ k \times l\ @\ m$ is a convolutional layer that contains m weight filters with the size of $k \times l$. A box represented by $max\ pool\ k \times k$ is a pooling layer with the size of $k \times k$, and fc is a fully-connected layer. A ReLU layer is followed for every convolutional and fully-connected layer.

C. Localization and Size Matching

In the test phase, an input image is resized to 80%, 100%, and 130% for considering various sizes of slab identification numbers in factory scenes, and patch images with the fixed size of 80×56 were collected from each resized image using a sliding window method with vertical 10-pixel or horizontal 8-pixel displacement. Collected patch images are classified by a trained DCNN, and probabilities that each patch is classified as a character patch are obtained. These probabilities are called character confidence scores. The horizontal projection profile of an image of character confidence scores was calculated, and 1-dimensional Non-Maximum Suppression (NMS) [9] was conducted to estimate vertical locations of identification numbers.

To accurately recognize a slab identification number, the size of characters in test patches should be similar to the fixed size of training patches (80×56). From a horizontal sequence of character confidence scores for each estimated vertical location in each resized image, indexes of the confidence scores above a threshold probability are obtained[1]. Character centers of an identification number in a resized image are estimated by calculating the center points for the sets of adjacent upper-threshold indexes. A center with a distance bigger than three times of 56 from the closest center is regarded as a center of a misclassified character, and it is removed. For sequences of confidence scores from similar relative positions in resized images, the length between the leftmost and rightmost centers of characters are compared to the value of 9 × 56. A sub-image for the sequence of confidence scores with the minimum difference is extracted from the corresponding resized image to recognize a slab identification number. The size of an extracted sub-image is 80 (height) × W (width), and W is the width of the corresponding resized image. The process for estimating the length between the leftmost and rightmost centers is illustrated in the Fig. 2.

D. Character Response Map Decoding

From a sub-image ($80 \times W$) for a slab identification number obtained from an appropriate resized image, patch images with the size of 80 × 56 are densely collected by using a sliding window method with a horizontal 2-pixel displacement, and these patches are classified with the use of the DCNN. The classification results for the patches from a sub-image are used to construct a character response map. The size of a response map is $11 \times [W/2]$, where $[W/2]$ is the number of patches collected from the sub-image, and each column of a response map represents a classification result for a patch. An element in the i^{th} row of a response map indicates the probability that the corresponding patch is classified as the i^{th} character. The $1^{st} \sim 10^{th}$ rows contain the probabilities

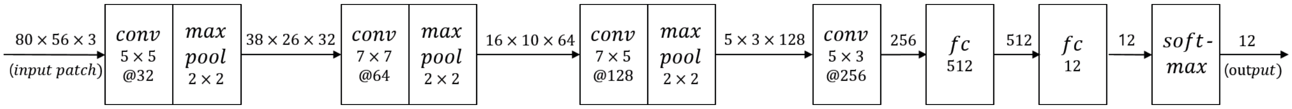


Fig. 1. An architecture of a DCNN.

[1] A heuristic probability of 0.3 was used for the threshold, but the performance of the proposed algorithm is not sensitive for the parameter.

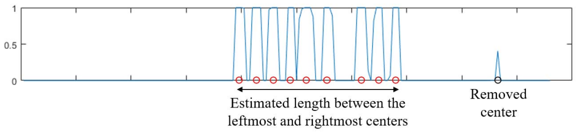
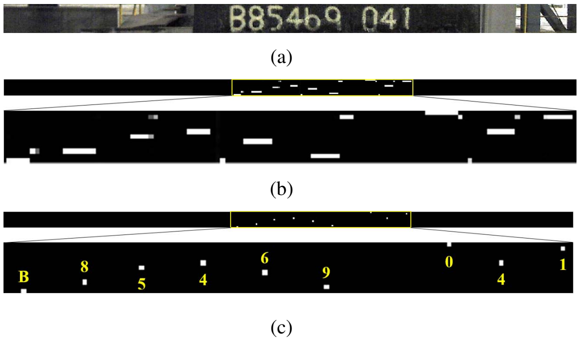


Fig. 2. A process for estimating the length between the leftmost and rightmost centers. Red circles indicate centers of characters in an idenficiation number.



(a)

(b)

(c)

Fig. 3. Response map decoding. (a) A sub-image from an appropriate resized image. (b) The character response map of the sub-image. (c) The result of 2-D NMS for the response map.

that the corresponding patch belongs to a numerical character in *0* ∼ *9*, and the 11^{th} row contains the probability for the alphabetic character *B*.

Character response maps are decoded or removed to transcribe slab identification numbers in a factory scene. Two-dimensional non-maximum suppression with the kernel size of $80 \times k$ is used to decode a character response map, and the parameter k is selected by the nearest upper integer of a quarter of the patch width. Each remaining point after the 2-D NMS procedure indicates a character candidate, and a response map with an output that has less than 9 points is removed. Several errors for the character-level classification occurred between *B* and *8* because of their similar shapes. At the end of the decoding process, *8* at the first character is exchanged to *B*, and *B*'s are exchanged to *8* except the first character. A slab identification number is transcribed by sequentially recording the remaining points in the form of characters, and Fig. 3 shows a procedure for response map decoding.

III. Experiment Result

The DCNN was trained with 78 million character patches and 52 million background patches for 45 epochs. The initial learning rate was 0.01, and it was declined by 1% for every epoch.

The proposed algorithm was tested on 426 slabs that is independent to the training data, and it yielded 99.77% of accuracy for the character-level classification and 94.84% of accuracy for the end-to-end recognition. The performance was measured by the number of correctly recognized identification numbers divided by the total number of slabs.

The localization performance of the previous algorithm [8] for the identical database was 94.9%, and its estimated end-to-end recognition accuracy is 92.91% with the use of our DCNN for the character classification. The performance improvement for the end-to-end recognition is estimated by 1.93% by using an integrated recognition strategy based on a deep learning methodology. Result images of our proposed algorithm are presented in Fig. 4.

TABLE I
Experimental result.

	Proposed	End-to-end recognition based on [8]
Accuracy (%)	94.84	92.91

IV. Discussion

In this section, current limitations of the proposed algorithm are discussed.

A. Epoch and Architecture of DCNN

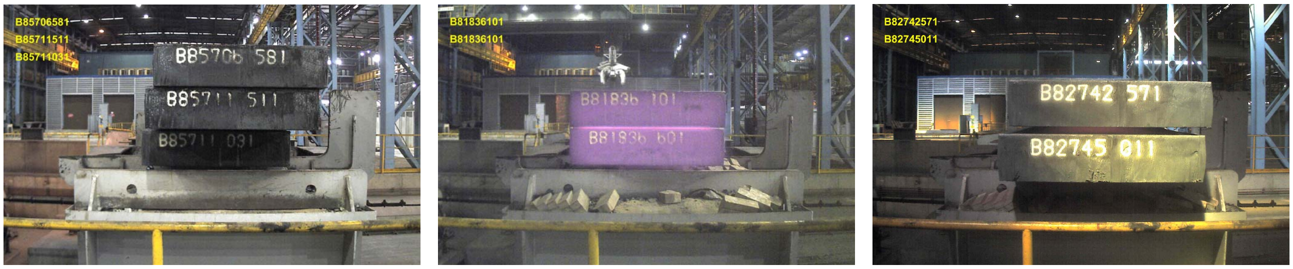
The training capacity of a DCNN architecture, an ability to train a complex model, is closely related to the number of layers and size of convolutional filters, and these contents should be designed with the consideration about available training data. In the field of machine learning, epoch represents the number of iteration paths for the all data, and the performance of a DCNN becomes generally improved or saturated as the number of epochs is increased for a sufficiently big data. However, the end-to-end recognition performance of the DCNN was fluctuated with respect to the number of epochs as shown in Fig. 5. It means the proposed architecture of DCNN is not optimized for the training data, and further researches are needed about the architecture of a DCNN. In the other point of view, this fluctuated performance implies that the performance of the proposed integrated end-to-end strategy can be improved with more nearly optimized structure of a DCNN.

B. Misclassification of Between-character Patches

In some cases, deep learning based methods yield fool results for untrained data that is not considered in the training phase [10]. In our training framework, there are two types of untrained data in the test phase: resized background patches and patches for between-character regions, and the second type of untrained data causes a problem for the recognition. Several failures for the recognition are arisen due to misclassifications of patches in between-character regions, and Fig 6 shows a failure case. Further researches are necessary for effectively decoding a character response map.

V. Conclusion

In this paper, an integrated end-to-end algorithm was proposed for recognizing slab identification numbers in factory scenes. The proposed algorithm was developed and evaluated in an actual industrial setting, and achieved 94.84% of accuracy for the end-to-end recognition. It was almost equal to the localization performance of the previous algorithm, and performance improvement for the end-to-end recognition



(a) (b) (c)

Fig. 4. Result images. Recognition results are marked at left-top corner in each factory scene. A result is marked with yellow color for a correctly recognized result or with red color for a failure case. Shadows are casted on the bottom slabs in (a). Reddish color is appeared on slabs in (b), and a different lighting condition cause a shining color for identification numbers in (c).

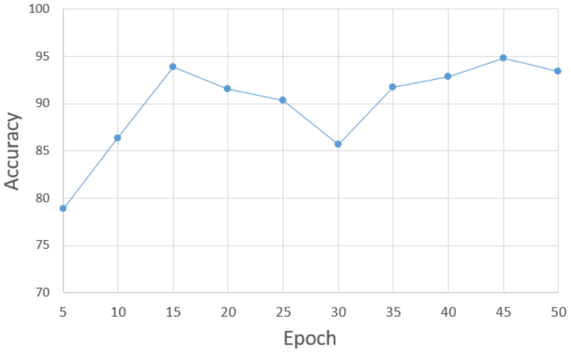
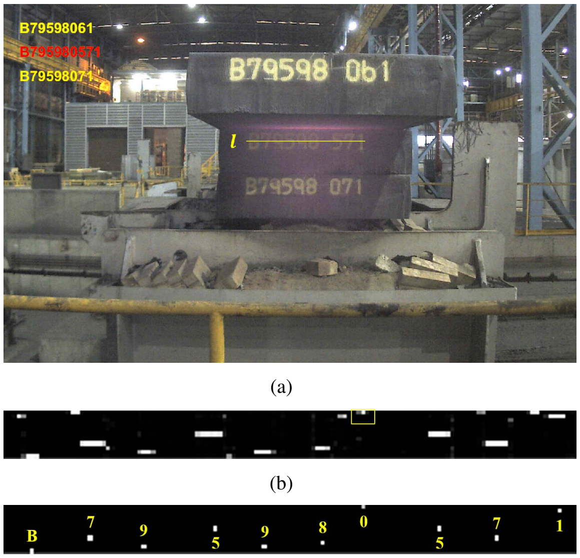


Fig. 5. Performance analysis for the number of training epochs.



(a)

(b)

(c)

Fig. 6. Incorrectly recognized result. (a) The slab idenficiation number of the middle slab is failed to recognize the product information. (b) A character response map along the horizontal line l in (a); the yellow box shows misclassifications of patches in a between-character region. (c) An incorrectly decoded result.

is estimated to 1.93%. Current limitations of the proposed algorithm will be overcome in future work.

Recently, numerous researches have been conducted about deep learning, but application-specific results are not sufficient especially in the industrial fields. This paper is certainly expected to become an important case-study with a deep convolutional neural network for an industrial application.

References

[1] L. Neumann and J. Matas, "Scene text localization and recognition with oriented stroke detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 97–104.

[2] A. Coates, B. Carpenter, C. Case, S. Satheesh, B. Suresh, T. Wang, D. J. Wu, and A. Y. Ng, "Text detection and character recognition in scene images with unsupervised feature learning," in *2011 International Conference on Document Analysis and Recognition*. IEEE, 2011, pp. 440–445.

[3] L. Neumann and J. Matas, "Real-time scene text localization and recognition," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 3538–3545.

[4] X.-C. Yin, X. Yin, K. Huang, and H.-W. Hao, "Robust text detection in natural scene images," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 5, pp. 970–983, 2014.

[5] H. Chen, S. S. Tsai, G. Schroth, D. M. Chen, R. Grzeszczuk, and B. Girod, "Robust text detection in natural images with edge-enhanced maximally stable extremal regions," in *Image Processing (ICIP), 2011 18th IEEE International Conference on*. IEEE, 2011, pp. 2609–2612.

[6] Y.-F. Pan, X. Hou, and C.-L. Liu, "Text localization in natural scene images based on conditional random field," in *2009 10th International Conference on Document Analysis and Recognition*. IEEE, 2009, pp. 6–10.

[7] Q. Ye and D. Doermann, "Text detection and recognition in imagery: A survey," *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 7, pp. 1480–1500, 2015.

[8] S. Choi, J. P. Yun, K. Koo, and S. W. Kim, "Localizing slab identification numbers in factory scene images," *Expert Systems with Applications*, vol. 39, no. 9, pp. 7621–7636, 2012.

[9] A. Neubeck and L. Van Gool, "Efficient non-maximum suppression," in *18th International Conference on Pattern Recognition (ICPR'06)*, vol. 3. IEEE, 2006, pp. 850–855.

[10] A. Nguyen, J. Yosinski, and J. Clune, "Deep neural networks are easily fooled: High confidence predictions for unrecognizable images," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2015, pp. 427–436.