ESANN 2019 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 24-26 April 2019, i6doc.com publ., ISBN 978-287-587-065-0.
Available from http://www.i6doc.com/en/.

A document detection technique using convolutional neural networks for optical character recognition systems

Lorand Dobai[1,2] and Mihai Teletin[1,2]

1- Lateral Inc., Cluj-Napoca, Romania
2- Babes-Bolyai University, Cluj-Napoca, Romania
{lorand.dobai,mihai.teletin}@lateral-inc.com

Abstract. An important part of an optical character recognition pipeline is the preprocessing step, whose purpose is to enhance the conditions under which the text extraction is later performed. In this paper, we present a novel deep learning based preprocessing method to jointly detect and deskew documents in digital images. Our work intends to improve the optical recognition performance, especially on frames which are skewed (slightly rotated) or have cluttered backgrounds. The proposed method achieves good document detection and deskewing results on a dataset of photos of cash receipts.

1 Introduction

Optical character recognition (*OCR*) is the task of transforming images of printed or handwritten text into machine encoded text. The main goal of *OCR* is to digitize printed texts so they can be electronically stored, manipulated, and displayed. Due to the high variational complexity of the problem the performance of these systems greatly relies on the quality of the preprocessing techniques employed. One important step is skew correction (deskewing), the process of aligning the document in order to make the lines of text as horizontally straight as possible.

We propose a novel preprocessing method based on document detection which uses deep learning and projective transformation. The method is using a *convolutional neural network* to detect the key points of the document, then uses these points for projecting the document into a rectangular shape. Moreover, we show that our method is capable of both detection and skew correction on document images. We build a lightweight model that can be used on a large variety of devices. Our method performs a single shot detection for finding the key points. To the best of our knowledge, such a method of preprocessing was not yet employed in the literature.

The remainder of this paper is organized as follows. In Section 2, we present related work concerning document preprocessing methods which are mainly based on *Hough transform*. Basics aspects regarding deep learning and computer vision are presented in Section 3. Afterwards, the machine learning methodology used is discussed in Section 4. The experimental part including the evaluation of the work is described in Section 5, while Section 6 presents the conclusions of the paper and directions for further work.

2 Literature review

Various work showed that the preprocessing step is critical in order to achieve good *OCR* performance [1]. However, the preprocessing part is still biased on classical com-

ESANN 2019 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 24-26 April 2019, i6doc.com publ., ISBN 978-287-587-065-0.
Available from http://www.i6doc.com/en/.

puter vision approaches such as *Hough transform* based methods [1]. Other methods are based on *nearest neighbours* and *projection profile* [2].

Hough transform is one of the most common methods used in the computer vision literature since most of the deskew methods are based on it. A fast skew detection and correction method was developed by Singh et al. in [1]. The method pipeline was divided in 3 main steps: a preprocessing stage that uses a form of block adjacency graph, *Hough transform* and rotation. Furthermore, in [3] a method called *image autorotation* is presented. The idea was to determine a line that supports the left margin of the text area. The slope of the line is used in order to define the page rotation angle. Other Hough based pipeline variants are also proposed in [4] and [5], focusing exclusively on the problem of deskewing scanned documents, context in which document localization is not a concern.

The review conducted in [2] showed that in terms of speed, the *nearest neighbour* technique achieves the fastest time. However, *nearest neighbour* performs poorly regarding the accuracy of the estimations. Another technique, *projection profile* gives the best angle estimation even though it takes the longest time to execute.

Xiong presented in [6] a fast method for document detection. It relies on classical computer vision methods such as: the *Canny edge operator*, *Hough transform*, and local maxima identification in the Hough coordinate space for predicting the 4 corners of the document which are then used to project the document into a deskewed form. The method militates for both speed and performance, being able to run smoothly on mobile devices.

The capacity of *convolutional neural network*s to predict bounding boxes of objects, by framing the object detection task as a regression problem has already been demonstrated with Overfeat [7]. In [8] a neural networks based approach was proposed for document localization. While the idea is similar to ours, there are some important differences. The method uses low resolution images and the detection is performed in a recursive manner while we militate for single shot localization on high resolution images. Moreover, the problem tackled in [8] is focused more on localization rather than deskewing.

3 Background

In this section we are presenting some basic concepts regarding deep learning. We are also going to cover some image processing aspects that are important for our study such as *projective transformation* and *skew detection and correction*.

3.1 Deep learning

From a regression perspective, deep learning can be simply viewed as finding an approximation $\widehat{f}$ of a function $f : R^n \rightarrow R^m$. The most popular optimization algorithm, *stochastic gradient descent* (*SGD*) is used to solve this optimization problem in a supervised learning manner by minimizing a loss function [9]. While various loss functions are widely used in literature, in this work we are using the mean squared error *MSE*, expressed as $MSE(y,\widehat{y}) = \frac{1}{2} \cdot \sum_{i=1}^{m} (\widehat{y}_i - y_i)^2$. In the formula, y represents the ground truth (i.e. the correct value for $f(x)$ for an instance X) while $\widehat{y}$ represents the predicted value for x (i.e. $\widehat{y} = \widehat{f}(x)$).

ESANN 2019 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 24-26 April 2019, i6doc.com publ., ISBN 978-287-587-065-0.
Available from http://www.i6doc.com/en/.

One particular architecture of neural networks is the *convolutional neural network* (CNN). A CNN is a specialized type of neural network that is capable to process data which has a known grid-like topology (e.g. images) [9].

3.2 Computer vision

Homogeneous coordinates or projective coordinates are a system of coordinates used in projective geometry, as Cartesian coordinates are used in Euclidean geometry. Homogeneous coordinates have a range of applications, including computer graphics, where they allow *projective transformations* to be easily represented by a matrix. A *projective transformation* maps lines to lines while it does not necessarily preserve parallelism and it can be expressed by an invertible 3x3 matrix in homogeneous coordinates.

The *skew* of a document refers to the orientation angle of the contained text. An excessive skew angle in any direction will lead to many problems during the analysis of the image, significantly affecting the performance of an *OCR* system. Many methods have been proposed to detect the angle [2].

4 The proposed approach

We introduce in this section our methodology for modelling and solving the problem of document skew detection and correction in a supervised learning manner.

Our skew detection and correction method differs from the existing similar methods discussed in Section 2 since it relays on a lightweight deep neural network. The model is applied for detecting 4 corners of the document, which are then used to map the document into a rectangular shape using a projective transformation. Given proper approximations of the corners, the projective transformation step is capable to accurately correct the skew and at the same time segment the document from the background. We are training a deep learning model which takes as input an image of a document and is capable to estimate the coordinates of the 4 corners. Hence, we build a model capable to preprocess images of cash receipts.

4.1 The proposed machine learning model

Since we are attempting to solve an image processing problem using machine learning, we choose to use a *convolutional neural network*. We are going to build our model based on a *MobileNet* [10] backbone. *MobileNet* is a lightweight deep neural network architecture suited for deployment on devices with small computational performance, such as smart phones.

The input of the model is represented by a photography of a cash receipt. The image is encoded as a *RGB* tensor denoted as x. Given x, the model computes the location of 4 points in the image domain representing the corners of the cash receipt. Each point is represented by a pair denoted as $(\widehat{y}_1, \widehat{y}_2)$ making our model return 8 values: $< \widehat{y}_{1,1}, \widehat{y}_{1,2}, \widehat{y}_{2,1}, \widehat{y}_{2,2}, \widehat{y}_{3,1}, \widehat{y}_{3,2}, \widehat{y}_{4,1}, \widehat{y}_{4,2} >$.

The architecture of the model is summarized as follow. The primal layers are represented by the convolutional layers of the base model, *MobileNet*. A global averaging pooling is applied on the output of the base model then the output is computed by 8 linear units.

ESANN 2019 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 24-26 April 2019, i6doc.com publ., ISBN 978-287-587-065-0. Available from http://www.i6doc.com/en/.

The mean squared error is a well known loss function that can be employed in our model. In this case, the loss is measuring the *Euclidean distance* between the predicted values ($\widehat{y} =< \widehat{y}_{1,1}, \widehat{y}_{1,2}, ..., \widehat{y}_{4,2} >$) and the ground truth ($y =< y_{1,1}, y_{1,2}, ..., y_{4,2} >$).

However, for our problem, only measuring the distance between the estimated point and the ground truth is not enough. To obtain good *skew* correction results, one has to consider the orientation of the points. Thus, the pairwise angles formed by the estimated points have to be as close as possible to the ground truth. In order to give prominence to how close is $\widehat{y}$ from y, we adapt the loss function by adding a new term, the *angular error*. We compute the angles between two vectors in 2D space, $v_1 = (v_{1,1}, v_{1,2})$ and $v_2 = (v_{2,1}, v_{2,2})$ as $\widehat{\theta}(v_1, v_2) = 180 - \frac{180 \cdot [atan2(v_{1,1}, v_{1,2}) - atan2(v_{2,1}, v_{2,2})]}{\pi}$, then the angles are normalized between 0 and 360 using (1).

$$\theta(v_1, v_2) = \begin{cases} \widehat{\theta}(v_1, v_2) - 360 & for\ \widehat{\theta}(v_1, v_2) > 360 \\ \widehat{\theta}(v_1, v_2) + 360 & for\ \widehat{\theta}(v_1, v_2) < 0 \end{cases} \tag{1}$$

We compute the four angles of the quadrilaterals determined by the ground truth points, respectively the predicted points. $\overrightarrow{Y_1} =< y_{2,1} - y_{1,1}, y_{2,2} - y_{1,2} >$; $\overrightarrow{Y_2} =< y_{3,1} - y_{2,1}, y_{3,2} - y_{2,2} >$; $\overrightarrow{Y_3} =< y_{4,1} - y_{3,1}, y_{4,2} - y_{3,2} >$; $\overrightarrow{Y_3} =< y_{1,1} - y_{4,1}, y_{1,2} - y_{4,2} >$. Then, the set of angles determined by our key points will be $U(y) = \{\theta(\overrightarrow{Y_1}, \overrightarrow{Y_2}), \theta(\overrightarrow{Y_2}, \overrightarrow{Y_3}), \theta(\overrightarrow{Y_3}, \overrightarrow{Y_4}), \theta(\overrightarrow{Y_4}, \overrightarrow{Y_1})\}$. Finally, the *angular error* is defined as the mean squared difference between the angles of the quadrilaterals which expresses the error between the angles of the predicted points and the ground truth, $AE(y, \widehat{y}) = \frac{1}{4} \sum_{i=1}^{4} (U_i(y) - U_i(\widehat{y}))^2$.

Thus, for optimizing the model, the proposed *loss function* which combines *mean squared error* and the *angular error* is defined as $E(y, \widehat{y}) = MSE(y, \widehat{y}) + \lambda \cdot AE(y, \widehat{y})$ where λ is a hyperparameter that controls the impact of the angular error.

5 Experimental evaluation

In this section we present the experiments conducted in order to assess the effectiveness of the proposed approach. It is composed of two main parts: *training* and *testing*. For developing the learning model we have employed the Keras implementation [11] while OpenCV [12] was used for the projective transformation.

5.1 Dataset

The dataset used in our experiment comprises a collection of photos representing various types of cash receipts collected from different sources. All the images are of the same high quality resolution (1920x1080). In order to decrease the complexity of the model for time efficiency reasons and without losing performance, all the images (and the corresponding labels) are down scaled to 480x270.

Let us denote the dataset as: $S = \{< x^{(1)}, y^{(1)} >, < x^{(2)}, y^{(2)} >, ..., < x^{(N)}, y^{(N)} >\}$. Each element of the set is composed of an input sample, represented by a photography (480x270) of a cash receipt denoted as x^i and its ground truth denoted as y^i. The ground truth is characterized by the 4 corners of the cash receipt: $y^{(i)} =< y^{(i)}_{1,1}, y^{(i)}_{1,2}, y^{(i)}_{2,1}, y^{(i)}_{2,2}, y^{(i)}_{3,1}, y^{(i)}_{3,2}, y^{(i)}_{4,1}, y^{(i)}_{4,2} >$, where $y^{(i)}_{j,1}$ is the x coordinate and $y^{(i)}_{j,2}$ is the y coordinate of the j^{th} key point of the sample $\forall 1 \leq i \leq N,\ \forall 1 \leq j \leq 4$. The key points are listed in a clockwise order, starting from the bottom left side of the document and ending with the

ESANN 2019 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 24-26 April 2019, i6doc.com publ., ISBN 978-287-587-065-0.
Available from http://www.i6doc.com/en/.

Figure 1: Test samples including the original image and the projection of the detected receipt. The key points were correctly marked even though the corner was obstructed.

bottom right. The input values are rescaled to $[-1,1]$ [10]. For our experiment, we use a dataset composed of 6000 entries.

For constructing the ground truth values we have developed a web application which was used to manually annotate the dataset. The application presents samples of photographies to the user which is asked to mark the 4 key points of the document. After marking the coordinates, the user is shown the transformed image obtained by applying the *projective transformation* into a rectangular shape using the points they placed. The user can further adapt the given estimations in order to obtain a good *skew correction*.

5.2 Experimental methodology and results

The *training* is performed on 90% of the dataset. We employ a distinct validation set formed of the remaining 10%. Training is performed using the Adam optimizer [13] with a learning rate of 0.001. A mini batch strategy is employed with a size of 16. The performance of the model denoted by the value of E introduced in Section 4.1 is measured on the validation set after the end of each epoch. The best performing versions of the model will be further used to report the results in the testing phase.

The obtained model is tested against a new collection of images, the testing set consisting of 700 images of cash receipts. The cash receipts come from different providers than those found in the training set and were designed to be very difficult. For each instance we detect the 4 key points and we report the *mean absolute error*, the *angular error* and the *absolute value* of the skew angle on the resulted projection. We perform three experiments: one version is using the classical MSE as loss ($\lambda = 0$) while the other two versions are using the proposed loss function ($\lambda \in \{1,5\}$). The obtained results are depicted in Table 1. We report the MAE, the Angular error and the mean of the absolute values of the skew angles. The experiments were repeated 10 times for each $\lambda \in \{0,1,5\}$ in order to compute the 95% confidence intervals (CI). Test samples including the detection and the projection are depicted in Figure 1. The reported metrics show that the best model in terms of skew correction was obtained using the model trained with the proposed loss function with $\lambda = 5$.

The average inference time per frame for our model is 109ms on a *OnePlus*5 mobile device, using the TFLite runtime (from TensorFlow 1.12) with 8 threads. Table 2 depicts the results obtained by comparing our method against related work. For an accurate comparison we run all algorithms on the same test set, computing the skew angle using the Hough transform over the obtained projections. The minimum, maximum, mean and standard deviation of the obtained Hough values along with their correspond-

ESANN 2019 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 24-26 April 2019, i6doc.com publ., ISBN 978-287-587-065-0. Available from http://www.i6doc.com/en/.

ing 95% CI are reported. The results show an overall better performance for our method compared to the approaches from [6] and [8].

Model	MAE	Angular error	Hough
MobileNet $\lambda = 0$	3.66 ± 0.07	4.87 ± 0.07	1.04 ± 0.01
MobileNet $\lambda = 1$	3.37 ± 0.04	4.05 ± 0.12	0.96 ± 0.01
MobileNet $\lambda = 5$	3.38 ± 0.05	3.22 ± 0.14	0.88 ± 0.01

Table 1: Results with 95% CIs.

Model	Min	Max	Mean	Std
MobileNet $\lambda = 0$	0.00 ± 0	9.61 ± 0.08	1.04 ± 0.01	1.16 ± 0.02
MobileNet $\lambda = 1$	0.00 ± 0	9.33 ± 0.07	0.96 ± 0.01	1.10 ± 0.02
MobileNet $\lambda = 5$	0.00 ± 0	9.62 ± 0.05	0.88 ± 0.01	2.20 ± 0.02
Xiong [6]	0.00	9.45	1.50	1.84
Javed and Shafait [8]	0.00	9.79	3.75	2.80

Table 2: Comparison to related work based on Hough values. 95% CIs are provided.

6 Conclusions and further work

We presented a new preprocessing technique effective for making images more accesible to *OCR* algorithms. It combines two steps: document detection and deskewing.

Further work will be performed on extending our method to be used for different kinds of documents. We plan to generalize the angular loss for n key points, suiting detection problems where the objects of interest appear in higher order polygonal forms.

Acknowledgement

The authors want to thank professor Gabriela Czibula from Babeş-Bolyai University, Romania for the given assistance and helpful suggestions.

References

[1] Chandan Singh, Nitin Bhatia, and Amandeep Kaur. Hough transform based fast skew detection and accurate skew correction methods. *Pattern Recognition*, 41(12):3528–3546, 2008.

[2] Arwa Al-Khatatneh, Sakinah Ali Pitchay, and Musab Al-qudah. A review of skew detection techniques for document. In *Modelling and Simulation (UKSim), 2015 17th UKSim-AMSS International Conference on*, pages 316–321. IEEE, 2015.

[3] Wojciech Bieniecki, Szymon Grabowski, and Wojciech Rozenberg. Image preprocessing for improving ocr accuracy. In *Perspective Technologies and Methods in MEMS Design, 2007. MEMSTECH 2007. International Conference on*, pages 75–80. IEEE, 2007.

[4] Riaz Ahmad, S Faisal Rashid, M Zeshan Afzal, Marcus Liwicki, Andreas Dengel, and Thomas Breuel. A novel skew detection and correction approach for scanned documents. In *DAS 2016, 12th Intl IAPR Workshop on Document Analysis Systems, At Santorini, Greece*, 2016.

[5] Abdelhak Boukharouba. A new algorithm for skew correction and baseline detection based on the randomized hough transform. *Journal of King Saud university-computer and information sciences*, 29(1):29–38, 2017.

[6] Ying Xiong. Fast and accurate document detection for scanning, August 2016.

[7] Pierre Sermanet, David Eigen, Xiang Zhang, Michaël Mathieu, Rob Fergus, and Yann LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv preprint arXiv:1312.6229*, 2013.

[8] Khurram Javed and Faisal Shafait. Real-time document localization in natural images by recursive application of a cnn. In *Document Analysis and Recognition (ICDAR), 2017 14th IAPR International Conference on*, volume 1, pages 105–110. IEEE, 2017.

[9] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.

[10] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[11] François Chollet et al. Keras, 2015.

[12] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[13] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.