ISSN 1870-4069

Deep Learning-based Methods for Face and Text Detection in Natural Images

Marco López-Sánchez,Oscar Chávez-Bosquez, Betania Hernández-Ocaña,
José Hernández-Torruco

Universidad Juárez Autónoma de Tabasco,
División Académica de Ciencias y Tecnologías de la Información,
Mexico

{marco.lopezsanchez, oscar.chavez,
betania.hernandez,jose.hernandezt}@ujat.mx

Abstract. The automatic detection of elements within an image has been the subject of numerous investigations in Computer vision. Detecting the objects making up an image and their relationship provides information that helps to interpret the scene's meaning. In this work, five methods based on Deep learning were evaluated, two for face detection and three for text detection. The methods for face detection are Dlib (Library for Machine Learning) and MTCNN (Multi-task cascaded convolutional neuronal networks). On the other hand, the evaluated methods for text detection are TesseractOCR, EasyOCR, and PaddleOCR. Results obtained with the evaluation indicate that the best face detection method was MTCNN and the best text detection method was EasyOCR. After analyzing the results, we propose a model based on MTCNN and EasyOCR to identify faces and texts in natural images simultaneously.

Keywords: Face detection, text detection, deep learning.

1 Introduction

The automatic detection of faces in natural images is one of the most studied topics in Computer vision [13]. Human faces are unique and cannot be reproduced, and they also provide information about human identity [25]. Detection is the first step for all facial analysis methods, such as facial recognition, face modeling, face verification, and face tracking. [19]. Facial detection is also used in the entertainment market (video games [26], virtual reality [6], and photo galleries [24]).

Regarding automatic text detection, several methods have been developed for text detection in natural images, becoming an active research field due to the growing demand for solutions to some artificial vision problems.

Detecting texts in natural scenes is a more challenging than detecting texts in scanned documents. Detecting the locations of the texts in the scene is

pp. 17–27; rec. 2022-05-03; acc. 2022-07-12

complicated because they are present in a scattered way, and in some cases, the appearance of the text makes it difficult to segment it.

In this work, two different Deep learning methods used for face detection are evaluated: (1) the Dlib [11], and (2) the MTCNN [10]. For text detection, three state-of-the-art methods are evaluated: (1) TesseractOCR [2], (2) EasyOCR [1], and (3) PaddleOCR [5]. For this evaluation, three subsets derived from the public data sets of Flickr8k [21], and COCO-Text [27] have been used.

The rest of the article is organized in the following order: Section 2 briefly reviews the background and state-of-the-art Deep learning-based methods used for face detection and text detection on natural images. Section 3 introduces the materials and methods. The experimental design is described in Section 4. Results and discussion are part of Section 5. Finally, conclusions are presented in Section 6.

2 State of the Art

Between the '70s and '80s, templates and measurements of geometric features were used to detect and recognize faces [17]. Early face detection efforts were primarily based on the traditional approach. The features were handcrafted from the image and introduced into a classifier to detect likely face regions. For this, two classic methods were used: the Histogram of Oriented Gradients (HOG) [4] and the HAAR Cascades classifier [28]. Despite the success of these methods, in recent years, models based on Deep learning have obtained outstanding results. In [3], they propose a face detector based on YOLOv3 [22], including a more accurate regression loss function and more appropriate anchor frames for the face. In [9], they propose a method based on Complete Discriminative Features (DCF) to improve face detection speed. This method uses a CNN that performs face detection directly on feature maps. Finally, Zhang et al.[32] proposed the FANet framework to build a detector that achieves high performance detecting faces with varied scales and features.

On the other side, traditional methods for text detection are primarily based on the discriminating characteristics of text areas within an image. These methods were divided into two approaches: component-based methods [8,16,12] and window-runner-based methods [18,14,29].

Deep learning methods for text detection have recently been used to achieve outstanding results. For example, in [15], a text detector called TextBoxes++ uses an end-to-end convolutional network that detects arbitrarily oriented scene text with high efficiency and accuracy. In [31], a novel text detector called TextField is designed to detect texts from irregular scenes; this detector was also trained with a fully convolutional neural network. This article [23] presents a model based on convolution neural networks to identify the language of the detected scene texts.

3 Materials and Methods

3.1 Face Detection Methods

Dlib It is a deep learning-based method created specifically for face detection in images. It is based on the histogram of oriented gradients (HOG) and convolutional neural networks (CNN). This model extracts facial reference points to calculate the orientation of a face in the scene [11]. It was trained with 68 facial reference points that provide information about the mouth, eyes, and nose.

MTCNN Acronym of *Multi-task cascaded convolutional neuronal networks*, it detects faces using a cascade of convolution neural networks divided into three stages: detect candidate face windows, discard candidates in which there are no faces, and identifies in which of the candidates a face exists [33].It works identifying the positions of five facial landmarks, one at each eye, another at the tip of the nose, and the remaining two at the corners of the lips.

3.2 Text Detection Methods

TesseractOCR It is an open-source text recognition engine[1]. It uses an LSTM neural network-based OCR engine and started as a research project in HP labs, using it in their line of scanners. Then, it was adopted by Google and made available to the public as an open source project. It supports various image formats such as PNG, JPEG and, TIFF, and can recognize more than 100 languages.

EasyOCR It is an open-source library used for text detection in images and supports more than 42 languages for detection purposes[2]. It has a default Deep learning architecture that uses three different types of neural networks [30].

PaddleOCR It is a framework that offers a series of pre-trained models with Recurrent neural networks (RNN) and CNNs[3]. It is based on the PaddlePaddle (*(PArallel Distributed Deep LEarning)*) framework. It is used for the detection, classification, and recognition of texts. It supports more than 80 languages.

3.3 Dataset

To carry out this research, we create 3 subsets of data from two different datasets:

Faces dataset This subset of data was used to compare the performance of face detection methods; thus, it consists of pictures including one or more faces. It is composed of 60 images that we selected from the Flickr8k dataset [4]. This dataset includes 245 faces distributed among 60 images.

[1] `https://tesseract-ocr.github.io`

[2] `https://github.com/JaidedAI/EasyOCR`

[3] `https://github.com/PaddlePaddle/PaddleOCR/tree/release/2.2`

[4] `https://www.kaggle.com/adityajn105/flickr8k?select=Images`

Text dataset The second subset is used to compare the performance of the text detection methods. It comprises 60 images including texts in different orientations, diverse sizes, and different fonts. This subset derives from the COCO-Text data set [27]. A total of 355 words are distributed among 60 images.

Face-text dataset This subset contains 40 images, only considering pictures where both faces and texts were found in the scene. These images were extracted from the public data set COCO-Text [27]. A total of 126 faces and 211 words are included in the 40 images.

3.4 Evaluation Metrics

Following evaluation metrics [20] compute the performance of the methods employing the following results:

- TP: True Positive is when the real value is 1 (True), and the predicted value is also 1 (True). It represents the recognized elements in the image (faces or words).
- FP: False Positive is when the real value is 0 (False), and the predicted value is 1 (True). It represents false identifications. It occurs when the detector identifies a region of the image as a face or text, but none of the elements are present.
- FN: False Negative is when the real value is 1 (True), and the predicted value is 0 (False). It represents the elements (faces or words) included in the image but not identified by the detector.

Precision: It is the number of items correctly identified as positive out of a total of items identified as positive.

$$precision = \frac{TP}{TP + FP}.$$

Recall: It is the proportion of positive cases correctly identified by the detector.

$$recall = \frac{TP}{TP + FN}.$$

F-Score: It combines the precision and recall measures to return a more general quality measure of the model. It is calculated as the harmonic mean of the metrics mentioned above.

$$F\text{-}Score = 2\frac{precision \cdot recall}{precision + recall}.$$

4 Experimental Design

Experiments were conducted using the Python programming language, including implementations of the methods for face detection (Dlib and MTCNN) and text detection (TesseractOCR, PaddleOCR, and EasyOCR).

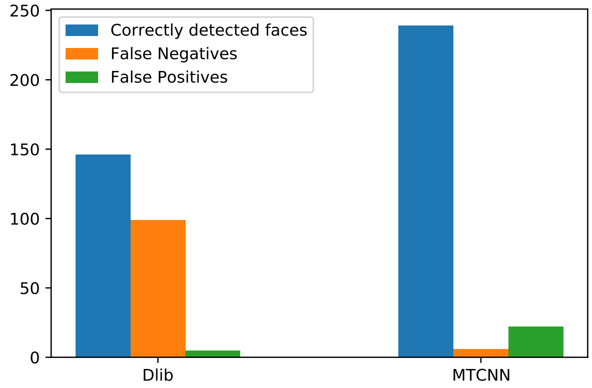


Fig. 1. Results obtained by Dlib and MTCNN in the Faces dataset.

Due the built-in support for implementing computer vision, we used the OpenCV [7] libraries. The following libraries were also used: `dlib 19.22.1`, `mtcnn 0.1.1`, `TesseractOCR 4.0.0`, `EasyOCR 1.6.2`, `PaddleOCR 2.6.0`, `Numpy 1.21.6` and `Matplotlib 3.2.2`. The default configuration of each detector was used.

Regarding the datasets, we performed manual labeling on each subset used in this work. Some faces found to be very blurred and difficult to identify by a human, so those images were not considered in the dataset. Also, incomplete or blurred words were not considered.

Three experiments were conducted to analyze the performance of the methods. In the first experiment, the methods for face detection were analyzed for all the elements of the Faces subset. In the second experiment, we analyze the three methods using all the elements of the Text dataset. In the third and last experiment, the proposed method was executed based on the methods that obtained the best performance. We use the Face-text dataset in this last experiment. All the experiments were conducted in Google Colab.

5 Results and Discussion

5.1 Face Detection Methods

Figure 1 shows the performance of the Dlib and MTCNN methods when evaluating the 60 images of the Faces dataset. We can notice that Dlib has a lower number of detected faces and a high number of False negatives, i.e., it could not detect 99 out of 245. On the other hand, MTCNN detected the most number of faces (239 out of 245), but it also detected 22 false positives (it detects faces where there are none).

Table 1. Face detection methods evaluation metrics.

Method	Precision	Recall	F-Score
Dlib	0.99	0.83	0.89
MTCNN	0.95	0.97	0.95

Table 2. Example of 3 images of the Faces dataset and corresponding results by Dlib and MTCNN.

Image	Total of faces	Dlib detection	MTCNN detection
	4	4	4
	4	2	4
	3	2	0

The best face detection method is highlighted in Table 1. Both Dlib and MTCNN methods were tested over the 60 images in the Faces dataset.

It should be noted that Dlib detected the fewest false positives, which is why it has the higher *precision*. However, it also detected the fewest true positives, which is why it has a lower recall. For this reason, the F-Score obtained by MTCNN is higher than that obtained by Dlib, indicating that MTCNN is a better detection method.

Table 2 shows 3 examples of the Faces dataset and the results obtained by the Dlib and MTCNN methods. We have included images with multiple faces, contrasting luminosity, and people in different scenes to test the face detection methods.

5.2 Text Detection Methods

Figure 2 shows the performance of the TesseractOCR, EasyOCR, and PaddleOCR methods when evaluating the 60 images of the Text dataset. We

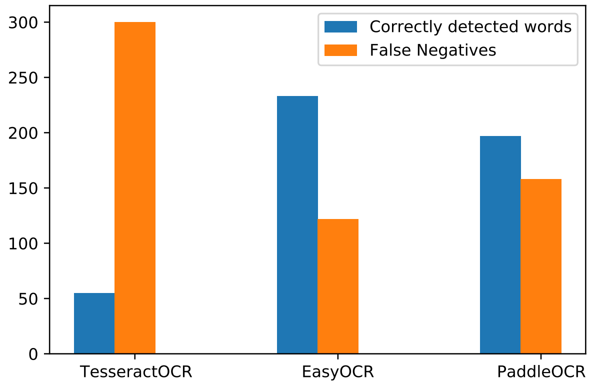


Fig. 2. Results obtained by TesseractOCR, EasyOCR, and PaddleOCR in the Faces dataset.

Table 3. Evaluation metrics results for the text detection methods.

Method	Precision	Recall	F-Score
TesseractOCR	1	0.54	0.70
EasyOCR	1	0.77	0.85
PaddleOCR	1	0.71	0.82

can notice that TesseractOCR has a lower performance, as it could only detect 55 out of 355 words. The best method was EasyOCR, detecting 233 out of 355 words, followed by PaddleOCR, with 197 out of 355 words.

The best text detection method is highlighted in Table 3. EasyOCR is the best of the three text detection methods evaluated with the Text dataset since the F-Score obtained exceeds the obtained by the other methods. None of the three methods detected false positives, so the result of the *precision* metric is 1. This means that 100 % of the words detected are found in the image, i.e., no false words are recognized in the scene. However, the *recall* metric indicates that EasyOCR indeed identifies more texts than the other methods, thus obtaining the highest *F-Score*.

Table 4 shows 3 samples of the Text dataset along with the results obtained by each text detection method. We notice that some words are not in a vertical orientation, yet EasyOCR obtained the best performance.

5.3 Proposed Face and Text Detection Method

The previous experiments allowed us to find the method with the best performance when identifying faces in natural images and the the best performance when identifying text in natural images to create a method to

Table 4. Example of 3 images of the Text dataset and corresponding results by TesseractOCR, EasyOCR, and PaddleOCR.

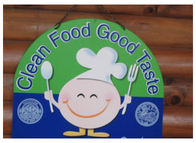
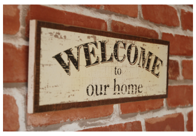
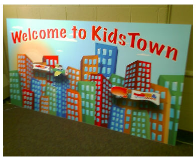
Image	**Total of words**	**TesseractOCR detection**	**EasyOCR detection**	**PaddleOCR detection**
	**4 words:** Clean Food Good Taste	0 words	2 words: - Food - Good	0 words
	**4 words:** WELCOME to our home	0 words	3 words: - to - our - home	3 words: - to - our - home
	**4 words:** Welcome to Kids Town	3 words: - Welcome - to - Kids	4 words: - Welcome - to - Kids - Town	0 words

Table 5. Result of the proposed method in the Face-text dataset.

Method	**Precision**	**Recall**	**F-Score**	**Global score**
MTCNN	1	0.71	0.82	0.84
EasyOCR	1	0.77	0.85	

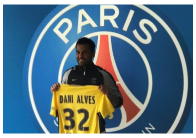
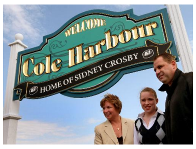
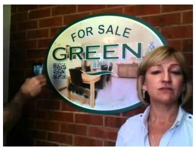
detect both faces and texts in images. Ee implemented the EasyOCR method for text detection, and the MTCNN method for face detection. We tested our model with the Face-text dataset, using a threshold of 0.7. Table 5 shows the results where the overall *F-Score* (the average of the two methods) is highlighted.

Table 6 shows 3 examples from the Face-text dataset and their corresponding results. The images in the subset contain faces in different positions; likewise, the text appears in different orientations and is presented in different font types and colors. These conditions result in a challenge for detection models. However, both methods obtained acceptable results.

6 Conclusion and Future Work

In this work, 5 Deep learning methods were evaluated: 2 for detecting faces in images and 3 for detecting texts in images. The performance of the face and text detection methods was compared using data subsets derived from the publicly available Flickr8K and COCO-Text datasets.

Table 6. Example of text and face detection using our proposal.

Image	Number of faces	Number of words	Faces detected	Words detected
	**1**	**4 words:** PARIS DANI ALVES 32	1	1 word: - ALVES
	**3**	**7 words:** Cole WELCOME Harbour HOME OF SYDNEY CROSBY	3	4 words: - OF - CROSBY - Cole - SIDNEY - HOME
	**1**	**3 words:** FOR SALE GREEN	1	3 words: - GREEN - SALE - FOR

MTCNN obtained the best overall performance in face detection. it has the highest *recall* than Dlib, although the latter obtains better *precision*. We choose MTCNN because it detects more faces per image than Dlib.

On the other hand, EasyOCR obtained the best results; it can detect slated words and text in curved orientations. However, the PaddleOCR method was the method that detected the most considerable amount of horizontally oriented words. Therefore, we opted for the EasyOCR method because it detects text in different orientations.

Finally, We proposed a custom method for face and text detection adopting the best methods in each category (face detection and text detection), intending to have an efficient model that recognizes both faces and texts with the best possible performance.

Future work will try to apply our model to different applications (counting people at events or public transport), apply recognition of detected faces, or even automatically evaluate the emotions of a given person by analyzing their gestures.

Acknowledgments. To the Consejo Nacional de Ciencia y Tecnología (CONACYT) for supporting the Doctoral program in Computer Science at the Universidad Juárez Autónoma de Tabasco.

References

1. Baek, Y., Lee, B., Han, D., Yun, S., Lee, H.: Character region awareness for text detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9365–9374 (2019)
2. Breuel, T. M., Ul-Hasan, A., Al-Azawi, M. A., Shafait, F.: High-performance OCR for printed English and Fraktur using LSTM networks. In: 2013 12th international conference on document analysis and recognition. pp. 683–687. IEEE (2013)
3. Chen, W., Huang, H., Peng, S., Zhou, C., Zhang, C.: Yolo-face: a real-time face detector. The Visual Computer, vol. 37, no. 4, pp. 805–813 (2021)
4. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05). vol. 1, pp. 886–893. Ieee (2005)
5. Du, Y., Li, C., Guo, R., Yin, X., Liu, W., Zhou, J., Bai, Y., Yu, Z., Yang, Y., Dang, Q., et al.: PP-OCR: A practical ultra lightweight OCR system. arXiv preprint arXiv:2009.09941, (2020)
6. Fuchter, S. K., Zucchi, S., Wortley, D.: Formative assessment of inquiry skills for responsible research and innovation using 3d virtual reality glasses and face recognition. In: Technology Enhanced Assessment: 21st International Conference, TEA 2018, Amsterdam, The Netherlands, December 10–11, 2018, Revised Selected Papers. vol. 1014, pp. 91. Springer (2019)
7. Gollapudi, S.: Learn computer vision using OpenCV. Springer (2019)
8. Greenhalgh, J., Mirmehdi, M.: Real-time detection and recognition of road traffic signs. IEEE transactions on intelligent transportation systems, vol. 13, no. 4, pp. 1498–1506 (2012)
9. Guo, G., Wang, H., Yan, Y., Zheng, J., Li, B.: A fast face detection method via convolutional neural network. Neurocomputing, vol. 395, pp. 128–137 (2020)
10. Jiang, B., Ren, Q., Dai, F., Xiong, J., Yang, J., Gui, G.: Multi-task cascaded convolutional neural networks for real-time dynamic face recognition method. In: International conference in communications, signal processing, and systems. pp. 59–66. Springer (2018)
11. King, D. E.: Dlib-ml: A machine learning toolkit. The Journal of Machine Learning Research, vol. 10, pp. 1755–1758 (2009)
12. Koo, H. I., Kim, D. H.: Scene text detection via connected component clustering and nontext filtering. IEEE transactions on image processing, vol. 22, no. 6, pp. 2296–2305 (2013)
13. Lal, M., Kumar, K., Arain, R. H., Maitlo, A., Ruk, S. A., Shaikh, H.: Study of face recognition techniques: A survey. International Journal of Advanced Computer Science and Applications, vol. 9, no. 6, pp. 42–49 (2018)
14. Lee, J.-J., Lee, P.-H., Lee, S.-W., Yuille, A., Koch, C.: Adaboost for text detection in natural scene. In: 2011 International conference on document analysis and recognition. pp. 429–434. IEEE (2011)
15. Liao, M., Shi, B., Bai, X.: Textboxes++: A single-shot oriented scene text detector. IEEE transactions on image processing, vol. 27, no. 8, pp. 3676–3690 (2018)
16. Mosleh, A., Bouguila, N., Hamza, A. B.: Image text detection using a bandlet-based edge detector and stroke width transform. In: BMVC. pp. 1–12 (2012)
17. Nixon, M.: Eye spacing measurement for facial recognition. In: Applications of digital image processing VIII. vol. 575, pp. 279–285. SPIE (1985)
18. Pan, Y.-F., Hou, X., Liu, C.-L.: A hybrid approach to detect and localize texts in natural scene images. IEEE transactions on image processing, vol. 20, no. 3, pp. 800–813 (2010)

19. Parekh, H. S., Thakore, D. G., Jaliya, U. K.: A survey on object detection and tracking methods. International Journal of Innovative Research in Computer and Communication Engineering, vol. 2, no. 2, pp. 2970–2978 (2014)
20. Powers, D. M.: Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. arXiv preprint arXiv:2010.16061, (2020)
21. Rashtchian, C., Young, P., Hodosh, M., Hockenmaier, J.: Collecting image annotations using Amazon's mechanical turk. In: Proceedings of the NAACL HLT 2010 workshop on creating speech and language data with Amazon's Mechanical Turk. pp. 139–147 (2010)
22. Redmon, J., Farhadi, A.: Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767, (2018)
23. Saha, S., Chakraborty, N., Kundu, S., Paul, S., Mollah, A. F., Basu, S., Sarkar, R.: Multi-lingual scene text detection and language identification. Pattern Recognition Letters, vol. 138, pp. 16–22 (2020)
24. Savchenko, A. V., Demochkin, K. V., Grechikhin, I. S.: Preference prediction based on a photo gallery analysis with scene recognition and object detection. Pattern Recognition, vol. 121, pp. 108248 (2022)
25. Simpson, E. A., Maylott, S. E., Leonard, K., Lazo, R. J., Jakobsen, K. V.: Face detection in infants and adults: Effects of orientation and color. Journal of experimental child psychology, vol. 186, pp. 17–32 (2019)
26. Solorzano Alcivar, N. I., Herrera Paltan, L. C., Lima Palacios, L. R., Paillacho Chiluiza, D. F., Paillacho Corredores, J. S.: Visual metrics for educational videogames linked to socially assistive robots in an inclusive education framework. In: Perspectives and Trends in Education and Technology, pp. 119–132. Springer (2022)
27. Veit, A., Matera, T., Neumann, L., Matas, J., Belongie, S.: COCO-text: Dataset and benchmark for text detection and recognition in natural images. arXiv preprint arXiv:1601.07140, (2016)
28. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001. vol. 1, pp. I–I. Ieee (2001)
29. Wang, K., Babenko, B., Belongie, S.: End-to-end scene text recognition. In: 2011 International conference on computer vision. pp. 1457–1464. IEEE (2011)
30. Xiao, Z., Liang, P.: Chinese sentiment analysis using bidirectional lstm with word embedding. In: International Conference on Cloud Computing and Security. pp. 601–610. Springer (2016)
31. Xu, Y., Wang, Y., Zhou, W., Wang, Y., Yang, Z., Bai, X.: Textfield: Learning a deep direction field for irregular scene text detection. IEEE Transactions on Image Processing, vol. 28, no. 11, pp. 5566–5579 (2019)
32. Zhang, J., Wu, X., Hoi, S. C., Zhu, J.: Feature agglomeration networks for single stage face detection. Neurocomputing, vol. 380, pp. 180–189 (2020)
33. Zhang, K., Zhang, Z., Li, Z., Qiao, Y.: Joint face detection and alignment using multitask cascaded convolutional networks. IEEE signal processing letters, vol. 23, no. 10, pp. 1499–1503 (2016)