Detecting Arbitrary Oriented Text in the Wild with a Visual Attention Model

Wenyi Huang† harrywy@gmail.com
Dafang He† duh188@psu.edu
Xiao Yang‡ xuy111@psu.edu
Zihan Zhou† zzhou@ist.psu.edu
Daniel Kifer‡ dkifer@cse.psu.edu
C. Lee Giles†‡ giles@ist.psu.edu

†Information Sciences and Technology, ‡Computer Science and Engineering
The Pennsylvania State University
University Park, PA 16802

ABSTRACT

Text embedded in images provides important semantic information about a scene and its content. Detecting text in an unconstrained environment is a challenging task because of the many fonts, sizes, backgrounds, and alignments of the characters. We present a novel attention model for detecting arbitrary oriented and curved scene text. Inspired by the attention mechanisms in the human visual system, our model utilizes a spatial glimpse network to processes the attended area and deploys a recurrent neural network that aggregates the information over time to determine the attention movement. Combining this with an off-the-shelf region proposal method, the model achieves the state-of-the-art performance on the highly cited ICDAR2013 dataset, and the MSRA-TD500 dataset which contains arbitrary oriented text.

Keywords

Scene Text Detection; Visual Attention; Deep Learning

1. INTRODUCTION

Text in natural scenes usually provides important semantic information about the scene and its content. A system that reads text in the wild will enable numerous multimedia applications such as improving automatic object recognition and image categorization [34], multimedia document indexing and retrieval, and assisting the visually impaired.

Although there have been numerous work on reading text in the wild [26, 27, 18, 11], the problem remains unsolved because of the insufficient accuracy in real-world applications. Especially, localizing text in natural scenes is extremely difficult because of the unconstrained fonts, sizes, backgrounds, inconsistent illumination, and occlusions. In addition, text in the wild is mostly captured with different orientations, perspective distortion and curved shapes. Most existing works on text detection have focused on horizontal or near-horizontal text. While a few works target localizing text with arbitrary orientations, perspectives, and skews [30, 6], these approaches still rely on hand-crafted features or rules for grouping oriented or skewed text.

Inspired by the presence of attention mechanisms in the visual system [20, 8] where humans recognize objects by moving attention to the next relevant parts of the object, we designed our text detection model to imitate this attention mechanism. Our attention-based model processes the scene text character by character in a sequential manner. Fig. 1 shows an example of how our model processes scene text. Specifically, our model is build up on a recurrent neural network (RNN) in combination with a convolutional neural network (CNN) [14]. At each step of the RNN, a CNN is deployed on a small area (the attended area) to perform local feature extraction. The extracted feature is input to the RNN, which aggregates the information over time, to determine the attended area for the next character. The model will stop when the last character of the word is encountered. Thus, the model follows the character sequence and localizes the word. Because of the attention mechanism, our model is able to localize arbitrarily oriented and skewed text.

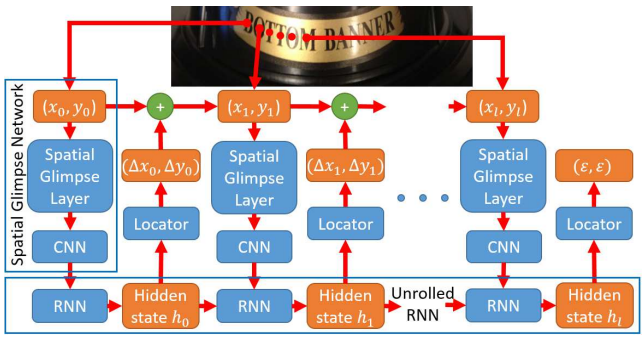

Figure 1: The architecture of the proposed attention model for text localization with an example of scene text detection.

For evaluation, our model was trained on a synthetically generated text in the wild dataset with ground truth locations of each character. We then conducted the experiments on two most cited datasets: the ICDAR2013 and the MSRA-TD500. We show that our model is capable of localizing arbitrarily oriented and skewed text without heuris-

MM '16, October 15 - 19, 2016, Amsterdam, Netherlands

ISBN 978-1-4503-3603-1/16/10...$15.00

DOI: http://dx.doi.org/10.1145/2964284.2967282

tically designed features and rules for grouping characters. Comparing with other baseline methods, our model achieves the state-of-the-art results. For oriented and skewed text, our method generates more accurate polygons as bounding boxes, some of them are even better than human labels.

2. RELATED WORK

2.1 Scene Text Detection

There are at least two major approaches in text localization. The first is sliding window based methods [7, 26, 27, 11] which slide a classifier over the entire image. The second is connected components methods, which group pixels into character regions using local properties such as color, gradient, intensity, stroke-width, etc. Pixels are group by algorithms such as Stroke Width Transform [9, 30], Extremal Regions [17, 18, 15], and Gradient Vector Flow [19]. The latter approaches have recently become more popular because they are usually more efficient and relatively insensitive to scale and orientation. However, most existing work has been focused on detecting horizontal or near-horizontal text. A few papers have considered detecting text with arbitrary orientations, perspectives, and skews [30, 6] based on hand-crafted features or rules for grouping oriented text.

2.2 Visual Attention

Attention mechanisms in human visual system has been proposed and studied in neuroscience [25, 20, 8]. Inspired by the mechanisms, various visual attention-based computer vision models have been proposed on tasks such as object recognition [1, 16, 2], object tracking [4], and image captioning [29]. In particular, our work extends the work of [16] and [2]. Mnih et al. [16] proposed a recurrent attention model for single object recognition which successfully learns the attention sequences on the MNIST dateset. Ba et al. [2] extends the recurrent attention model to recognize multi-digit on the SVHN dataset. The model is designed to recognize one big object (house number) with multiple components (multiple digits). The attention strategies are learnt based on a Cartesian coordinate that is centered at the middle of the input image. A hyper-parameter ratio that is used to convert unit width in the coordinate system to the number of pixels, however, limits the scalability of these models.

Different from these models, our model is relatively insensitive to the scale of text and the size of the image. In addition, instead of using reinforcement learning, our model is trained on synthetic text in the wild images, where ground truth locations are used for learning attention movement.

3. ATTENTION MODEL FOR TEXT LOCALIZATION

We approach the text localization task as a sequential process of attention pursuit. As in Fig. 1, at each step, a spatial glimpse network observes a limited area. It extracts the local information of the attended character area with different sizes and resolutions. Since the scene environment is partially observed by the spatial glimpse network, a recurrent neural network is deployed on top to aggregate the information over time. The integrated feature of the RNN is used by a locator network to determine where the model should attend to at the next step.

3.1 Model Components

Spatial Glimpse Network.

When a human recognize objects, the focus of our attention is within a small area. Meanwhile, we also maintain a blurry sense of the surrounding environment [25]. The spatial glimpse network is designed to imitate this visual attention mechanism. It is composed of two parts: a spatial glimpse layer [16] and a multi-column CNN.

Given an input image and a focusing coordinate (x, y), the spatial glimpse layer crops d image patches centered at (x, y) with multiple size and resolutions. The first image patch is $w \times w$ with the original resolution. This patch is considered as the focusing area where the model "sees" the most clearly. Each successive image patch is cropped twice the size of its previous image patch size then re-scaled to $w \times w$. Thus the model catches the surrounding environments of the focusing area. Fig. 2 shows an example of the input and the output of a glimpse layer with $d = 3$ and $w = 32$.

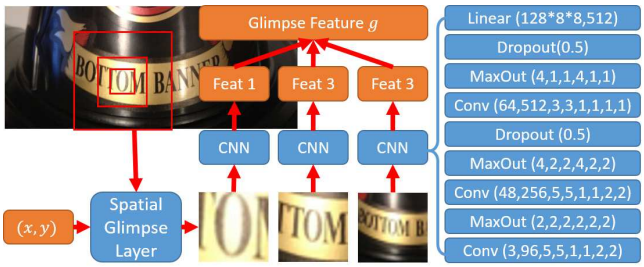

Figure 2: Input and output of a spatial glimpse layer with $d = 3$ and $w = 32$ and the details of the glimpse network.

The d glimpse patches are then input to a d-column convolutional neural network for feature extraction. Each column is a deep CNN and these CNNs do not share parameters. The output feature vectors from these d columns are concatenated as one feature vector which we refer to as the glimpse feature g. Details of the glimpse network and the structures of the CNNs are shown in Fig. 2.

Recurrent Neural Network.

In a cognitive visual attention mechanism, our brain aggregates aggravates the previously captured information over time to comprehend the whole picture of the environment [20, 8]. The integrated information helps us to determine the next location to move our attention to. Our attention model for text localization is designed to process words character by character, i.e. in a sequential manner. The location of the next character is determined by the sequence of characters that appears before. We use an RNN to aggregate the information extracted by the spatial glimpse network. At each step, the glimpse feature is input to the RNN. Meanwhile, the previous hidden state of the RNN is also the input to the current step. In our model, we use a simple RNN:

$$h_t = \text{ReLU}(W_{ih} \cdot g_t + W_{hh} \cdot h_{t-1} + b_r) \quad (1)$$

where h_t and h_{t-1} is the hidden state of the RNN at step t and $t-1$ respectively, g_t is the glimpse feature extracted by the spatial glimpse network at step t, W_{ih} is the weighting matrix from input to hidden, W_{hh} is the weighting matrix from hidden to hidden. We use the rectified linear unit (ReLU) as the activation function for the RNN.

Locator Network.

The hidden state of the RNN h_t is used to predict where to deploy the spatial glimpse network for the $t+1$ step. A locator network takes h_t as input and predicts where to attend to by outputing the movement offset (Δx_t,Δy_t). It consists of a fully connected layer that projects the hidden state h_t to a 2-dimensional vector, a tanh activation layer that rescales the numbers to [-1, 1] and a scaling layer that multiplies the vector by the maximal step length s_l.

$$(\Delta x_t, \Delta y_t) = s_l \tanh(W_l \cdot h_t + b_l) \quad (2)$$

and the next location is computed by:

$$(x_{t+1}, y_{t+1}) = (x_t, y_y) + (\Delta x_t, \Delta y_t) \quad (3)$$

Ideally, a well-trained model will move the attention to the location of the successive characters in the text.

3.2 Training

The training criterion is to minimize the Euclidian distance between the predicted location and ground truth location of each character in the word. The parameters of the glimpse network, the recurrent network, and the locator network are tuned by the loss function:

$$Loss_t = \frac{1}{2}\left[(x_t - \hat{x}_t)^2 + (y_t - \hat{y}_t)^2\right] \qquad \forall t \in [1, l] \quad (4)$$

where (x_t, y_t) is the predicted location at step t and $(\hat{x}_t, \hat{y}_t)$ is the ground truth center of the t^{th} character. For a word of length l, there will be $l-1$ backpropagation through time.

Curriculum Learning.

Since the predicted location at step t depends on all the previous moving strategies, the error will accumulate over steps. It will be meaningless if a prediction is made based on a wrong glimpse of a no-text area (i.e. the previous predicted attention area is far from the ground truth location). To solve this problem, we adapted the curriculum learning strategy [5] that we start with easy examples: The model starts training with data samples of sequence length 2. The sequence length increases as the average loss drops to a acceptable range (averaged Euclidian loss less than 9.00, i.e. 3 pixels away from ground truth). During the attention moving process, we will stop moving forward and start backpropagation when the current predicted center is far from the ground truth (i.e. larger than 3 pixels).

Two models.

Our attention model requires an initial location of a character in the text. However, there is no guarantee that the region proposal methods can always capture the first character of a word. To make our model robust, we enrich the training sequences to not only start from the very first character of a word, but also start from each character in the word. In addition, we trained two models that learn to localize text from left to right and right to left respectively. Given a proposed region, we apply the two models to localize the left part and the right part simultaneously.

3.3 Synthetic Data Generation

In order to train the attention model, we need a text in the wild dataset with ground truth locations of each character. However, existing datasets that meet the requirement contain too few images; also the variation of text orientation is limited. Considering the inaccuracy and the cost of human labeling, it's impractical to manually create a large training dataset that meet the requirement. Thus, we create our own synthetically generated text in the wild images with ground truth locations. The data generation process is similar to [10]. First, a font is randomly chosen from over 1,400 Google Fonts to generate the foreground text. Then several rendering effects, such as border/shadow rendering, rotation, perspective transformation, and Gaussian noise, are added to the generated images with random parameters. Based on the glyph metrics of each individual character as well as the kerning information of the font, we are able to calculate the exact bounding box and the center location of each letter. In the last step, we remove the text areas in ICDAR2013 and MSRA-TD500 training images, and blend the generated text into these natural scenes images.

All words are randomly selected from a pre-defined dictionary which contains over 90,000 common English words. Each generated image has a range of 1 to 3 lines of text with each line contains 1 to 3 words. Fig. 3 shows some examples of the synthetic text in the wild images.

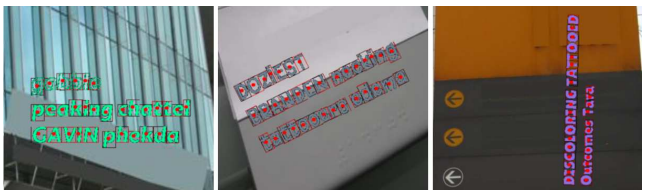

Figure 3: Examples of synthetic text in the wild images with bounding boxes and character centers.

4. TEXT LOCALIZATION PIPELINE

A full text localization pipeline involves three steps:

Region Proposal. We utilize an off-the-shelf region proposal method, Extremal Regions (ER) [18], to generate candidate character regions. The output regions of ER are then filtered by a CNN text/no-text classifier that we trained using the same architecture as in [11]. After filtering, we will have a set of proposed regions.

Localization with Attention Model. For each proposed region, we rescale the image so that the size of the region fits in $w \times w$. Then the left-to-right and the right-to-left models are applied to predicted the attended area of each character in the word. If the center of a predicted attended area by the model is within 3 pixels of a proposed region's center, we will remove the proposed region so that we do not have multiple detections of a same word. After this step, we will have several text line candidates.

Text line filtering. To remove the false positive text line candidates, we again use the text/no-text classifier to slide through the text lines. We will filter out a text line candidate if less than half of the text line regions are classified as text.

5. EXPERIMENTS

We set the number of cropped image patches d=3, the focusing area size w=32, and the maximal step length s_l=64. 30,000 synthetic images are generated with 90,000 most frequently used English words for training. For optimization, all model components are tuned with rmsprop [24] with batches of size 50. The initial learning rate is set to 0.005 and we halved the learning rate every 20 epoches. Our model is trained and tested on a single NVIDIA Tesla K40 GPU.

5.1 Dataset

We measure the performance of the proposed model on the ICDAR2013 [13] and the MSRA Text Detection 500 (MSRA-TD500) [30] datasets. ICDAR2013 mainly consists of horizontal text. The testing set contains 233 images. We evaluate our result on the ICDAR online evaluation system[1]. MSRA-TD500 is a multi-orientation text dataset containing text in both Chinese and English. The testing set consists of 200 images. We follow the evaluation protocols as in [30] and apply DetEval [28] to calculate the Precision, Recall and F-measure of our detection result.

5.2 Results

We first test our model on the ICDAR2013 dataset, which is the most cited horizontal text detection dataset. As in Table 1, the proposed method achieves 88%, 72%, 79% in Precision, Recall, and F-measure. Compared with the most recent methods that are only designed for horizontal text, our model achieve the best precision as the state-of-the-art method. The recall is a bit lower because we used an off-the-shelf region proposal method without fine-tuning.

	Precision	Recall	F-measure
Neumann and Matas [18]	73	65	69
Shi et al. [22]	83	63	72
Bai et al. [3]	79	68	73
Zamberletti et al. [32]	86	70	77
Tian et al. [23]	85	**76**	**80**
Zhang et al. [33]	**88**	74	**80**
Our model	**88**	72	79

Table 1: Localization performances on ICDAR2013(%).

We then experiment on the MSRA-TD500 dataset to validate our model's capability in localizing arbitrary oriented and skewed text in the wild. For each detection, our model outputs a sequence of centers. The final bounding box of a text line is created to be a polygon which is the union of all focusing areas in the sequence. Table 2 compares the results of our model with other baseline methods. The results show that the proposed method outperforms most of the other baselines in precision and F-measure. Compared to [31], our method does not sacrifice the recall to get a higher precision. With the same highest recall as in [21], our method achieves a much better precision result.

	Precision	Recall	F-measure
Chen et al. [7]	5	5	5
Epshtein et al. [9]	25	25	25
Yao et al. [30]	63	63	63
Risnumawan et al. [21]	70	**68**	69
Yin et al. [31]	**81**	63	**71**
Kang et al. [12]	71	62	66
Our model	74	**68**	**71**

Table 2: Localization performances on MSRA-TD500(%).

Besides the quantitative results, we also show the detection examples of some challenging cases from the MSRA-TD500 in Fig. 4. The yellow boxes indicate the ground truth. The red ones and the green ones are generated from our model, where a green polygon means a hit and a red polygon indicates a false positive detection. It is important to note that although our models are trained on synthetic dataset that only contains English words, the model shows the ability to localize Chinese characters as well.

[1]http://rrc.cvc.uab.es/

(a) Succeed cases of the proposed method

(b) Failed cases of the proposed method.

(c) Cases we failed following the evaluation protocol. However, our proposed model generates better bounding boxes.

Figure 4: Loalization results on MSRA-TD500 test set.

In Fig. 4a, we show that our proposed model is able to localize text of arbitrary orientation, perspective distortion, different scales, and inconsistent illumination. We also show that our generated polygons are tighter and more accurate than the ground truth boxes. Fig. 4b shows several failed cases of the proposed model, most of the failures are due to the fact that region proposal method, ER, sometimes cannot capture an entire Chinese character. Thus we would expect higher precision and recall if better region proposal methods are used. Fig. 4c shows some cases that our model failed the test following the evaluation protocol. However, we argue that in these cases, our method generates more accurate polygon bounding boxes than the human labeled ground truth. For example, in the first figure, "WELCOME" is broken into pieces, whereas "COME" and "TO EC" is merged as one box in the ground truth. Our model successfully detected "WELCOME" as one text region. In the second figure, our detection for the skewed text "AZONA" is much more accurate than the human labeled box. Our model's detections on the Chinese words in the last figure are more accurate than the loose boxes provided by the ground truth. In sum, by showing these detection results, we demonstrate that our proposed model generates more accurate polygons as bounding boxes for oriented and skewed text. Some of them are even better than human labels.

6. CONCLUSIONS

We propose a novel attention model for text detection in the natural scenes. The visual attention mechanism, implemented with CNNs and RNNs, provides a natural solution for localizing arbitrary oriented and skewed text. Evaluation results on two benchmark datasets show that our model generates more accurate bounding boxes for both horizontal and oriented text. For future work, we are going to explore our model's ability in the word recognition task.

Acknowledgments

This work was funded by NSF grant CCF-1317560 and a GPU donation by NVIDIA.

7. REFERENCES

[1] B. Alexe, N. Heess, Y. W. Teh, and V. Ferrari. Searching for objects driven by context. In *Proc. of NIPS'12*, pages 881–889, 2012.

[2] J. Ba, V. Mnih, and K. Kavukcuoglu. Multiple object recognition with visual attention. In *Proc. of ICLR'15*, 2015.

[3] B. Bai, F. Yin, and C. L. Liu. Scene text localization using gradient local correlation. In *Proc. of ICDAR'13*, pages 1380–1384. IEEE, 2013.

[4] L. Bazzani, N. de Freitas, H. Larochelle, V. Murino, and J.-A. Ting. Learning attentional policies for object tracking and recognition in video with deep networks. In *Proc. of ICML'11*, pages 937–944. ACM, 2011.

[5] Y. Bengio, J. Louradour, R. Collobert, and J. Weston. Curriculum learning. In *Proc. of ICML'09*, pages 41–48. ACM, 2009.

[6] M. Bušta, T. Drtina, D. Helekal, L. Neumann, and J. Matas. Efficient character skew rectification in scene text images. In *Proc of ACCV'14 Workshops*, pages 134–146. Springer, 2014.

[7] X. Chen and A. L. Yuille. Detecting and reading text in natural scenes. In *Proc. of CVPR'04*, pages 366–373. IEEE Computer Society, 2004.

[8] M. Corbetta and G. L. Shulman. Control of goal-directed and stimulus-driven attention in the brain. *Nature reviews neuroscience*, 3(3):201–215, 2002.

[9] B. Epshtein, E. Ofek, and Y. Wexler. Detecting text in natural scenes with stroke width transform. In *Proc. of CVPR'10*, pages 2963–2970. IEEE, 2010.

[10] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman. Synthetic data and artificial neural networks for natural scene text recognition. In *Workshop on Deep Learning, NIPS*, 2014.

[11] M. Jaderberg, A. Vedaldi, and A. Zisserman. Deep features for text spotting. In *Proc. of ECCV 2014*, pages 512–528. Springer, 2014.

[12] L. Kang, Y. Li, and D. Doermann. Orientation robust text line detection in natural images. In *Proc. of CVPR'14*, pages 4034–4041. IEEE, 2014.

[13] D. Karatzas, L. Gomez-Bigorda, A. Nicolaou, S. Ghosh, A. Bagdanov, M. Iwamura, J. Matas, L. Neumann, V. R. Chandrasekhar, S. Lu, et al. Icdar 2015 competition on robust reading. In *Proc. of ICDAR'15*, pages 1156–1160. IEEE, 2015.

[14] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. In *Intelligent Signal Processing*, pages 306–351. IEEE Press, 2001.

[15] J. Mao, H. Li, W. Zhou, S. Yan, and Q. Tian. Scale based region growing for scene text detection. In *Proc, of ACM MM'13*, pages 1007–1016. ACM, 2013.

[16] V. Mnih, N. Heess, A. Graves, et al. Recurrent models of visual attention. In *Proc. of NIPS'14*, pages 2204–2212, 2014.

[17] L. Neumann and J. Matas. A method for text localization and recognition in real-world images. In *Proc. of ACCV'10*, pages 770–783. Springer, 2010.

[18] L. Neumann and J. Matas. Real-time scene text localization and recognition. In *Proc. of CVPR'12*, pages 3538–3545. IEEE, 2012.

[19] T. Q. Phan, P. Shivakumara, and C. L. Tan. Detecting text in the real world. In *Proc. of ACM MM'12*, pages 765–768. ACM, 2012.

[20] R. A. Rensink. The dynamic representation of scenes. *Visual cognition*, 7(1-3):17–42, 2000.

[21] A. Risnumawan, P. Shivakumara, C. S. Chan, and C. L. Tan. A robust arbitrary text detection system for natural scene images. *Expert Systems with Applications*, 41(18):8027–8048, 2014.

[22] C. Shi, C. Wang, B. Xiao, Y. Zhang, and S. Gao. Scene text detection using graph model built upon maximally stable extremal regions. *Pattern recognition letters*, 34(2):107–116, 2013.

[23] S. Tian, Y. Pan, C. Huang, S. Lu, K. Yu, and C. Lim Tan. Text flow: A unified text detection system in natural scene images. In *Proc. of ICCV'15*, pages 4651–4659, 2015.

[24] T. Tieleman and G. Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural Networks for Machine Learning*, 4:2, 2012.

[25] H. von Helmholtz and J. P. C. Southall. *Treatise on Physiological Optics: Translated from the 3rd German Ed.* Optical Society of America, 1925.

[26] K. Wang, B. Babenko, and S. Belongie. End-to-end scene text recognition. In *Proc. of ICCV'11*, pages 1457–1464. IEEE, 2011.

[27] T. Wang, D. J. Wu, A. Coates, and A. Y. Ng. End-to-end text recognition with convolutional neural networks. In *Proc. of ICPR'12*, pages 3304–3308. IEEE, 2012.

[28] C. Wolf and J.-M. Jolion. Object count/area graphs for the evaluation of object detection and segmentation algorithms. *IJDAR*, 8(4):280–296, 2006.

[29] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhutdinov, R. Zemel, and Y. Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *Proc. of ICML'15*, 2015.

[30] C. Yao, X. Bai, W. Liu, Y. Ma, and Z. Tu. Detecting texts of arbitrary orientations in natural images. In *Proc. of CVPR'12*, pages 1083–1090. IEEE, 2012.

[31] X.-C. Yin, W.-Y. Pei, J. Zhang, and H.-W. Hao. Multi-orientation scene text detection with adaptive clustering. *IEEE Transactions on PAMI*, 37(9):1930–1937, 2015.

[32] A. Zamberletti, L. Noce, and I. Gallo. Text localization based on fast feature pyramids and multi-resolution maximally stable extremal regions. In *Proc. of ACCV'14*, pages 91–105. Springer, 2014.

[33] Z. Zhang, W. Shen, C. Yao, and X. Bai. Symmetry-based text line detection in natural scenes. In *Proc. of CVPR'15*, June 2015.

[34] Q. Zhu, M.-C. Yeh, and K.-T. Cheng. Multimodal fusion using learned text concepts for image categorization. In *Proc. of ACM MM'06*, pages 211–220. ACM, 2006.