Learning to Read Irregular Text with Attention Mechanisms

Xiao Yang, Dafang He, Zihan Zhou, Daniel Kifer, C. Lee Giles
The Pennsylvania State University, University Park, PA 16802, USA
{xuy111, duh188}@psu.edu, zzhou@ist.psu.edu, dkifer@cse.psu.edu, giles@ist.psu.edu

Abstract

We present a robust end-to-end neural-based model to attentively recognize text in natural images. Particularly, we focus on accurately identifying irregular (perspectively distorted or curved) text, which has not been well addressed in the previous literature. Previous research on text reading often works with regular (horizontal and frontal) text and does not adequately generalize to processing text with perspective distortion or curving effects. Our work proposes to overcome this difficulty by introducing two learning components: (1) an auxiliary dense character detection task that helps to learn text specific visual patterns, (2) an alignment loss that provides guidance to the training of an attention model. We show with experiments that these two components are crucial for achieving fast convergence and high classification accuracy for irregular text recognition. Our model outperforms previous work on two irregular-text datasets: SVT-Perspective and CUTE80, and is also highly-competitive on several regular-text datasets containing primarily horizontal and frontal text.

1 Introduction

Reading text from natural images is a challenging problem. The rich semantic information carried by the text is useful for many applications such as navigation, traffic sign reading for autonomous driving, and assistive technologies for the visually impaired. While reading text in scanned documents has been extensively studied and many production quality Optical Character Recognition (OCR) systems exist, reading text in natural images remains a difficult task. The imperfect imagery conditions in natural images, such as low resolution, blurring, and challenging perspective distortions have limited computers from accurately reading text in the wild.

A text reading system often consists of two parts: 1) scene text detection that localizes each word in natural images and 2) scene text recognition that takes a cropped image of a single word and outputs the depicted text. This work focuses on improving the second part. In particular, we find most existing studies consider horizontal and frontal text (referred to as *regular* text by Shi et al. [2016b]). These systems are not readily generalizable to processing *irregular* (perspectively distorted or curved) text. However, irregular text is pervasive in natural images. As shown in Figure 1, text captured by a side-view camera suffers from perspective distortion and text on bottles, products, or shop signs may have curved character placement. Therefore, developing a robust model to read both regular and irregular text is important for real world problems.

Figure 1: Examples of irregular (perspectively distorted or curved) text in natural images.

Here, we present an end-to-end, deep neural-based model that can accurately read irregular text. Inspired by the attention mechanism of human vision system and its analogy in several vision tasks, our model first learns high-level visual representations using a deep convolutional neural network (CNN), then attentively recognize sequence of characters with a recurrent neural network (RNN).

A related work to ours is Lee and Osindero [2016] who proposed a recursive recurrent net with attention modeling (R^2AM) for regular text recognition. But their approach is not directly applicable to handle irregular text reading. We observe that irregular character placement in rotated or curved text significantly increase the difficulty of neural nets training. To address this problem, we first introduce an auxiliary dense character detection task. This task encourages the learning of visual representations, by a fully convolutional network (FCN), that are favorable to the text patterns. Second, we propose an alignment loss to regularize the estimated attention at each time-step. Finally, we use a coordinate map as a second input to enforce spatial-awareness, which is helpful for the movement of attention. The architecture of our end-to-end model is illustrated in Figure 2. To train the proposed model, we generate a large-scale synthetic dataset containing perspectively distorted and curved scene text. Character-level bounding box annotations are also provided in addition to word annotations.

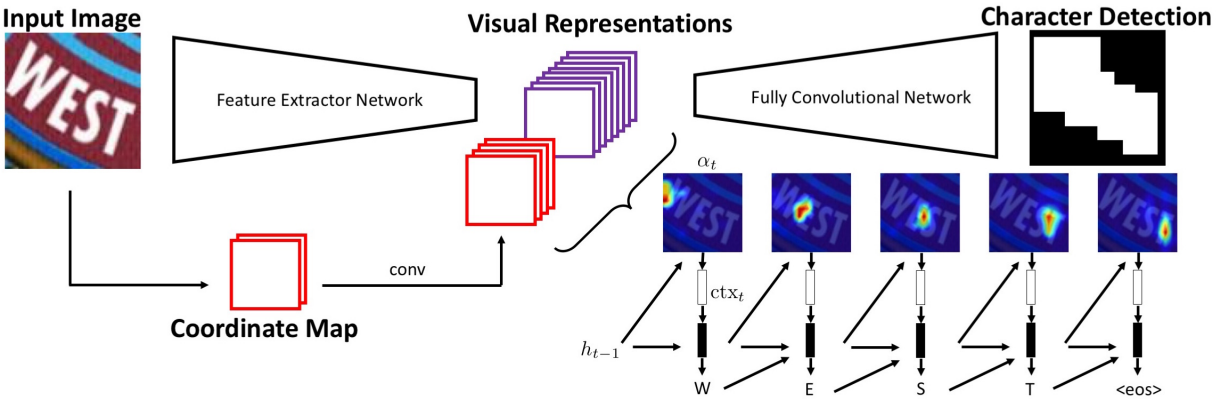

Figure 2: Our architecture consists of three components: 1) a convolutional feature extractor network f; 2) a fully-convolutional neural network g for dense character detection and 3) a recurrent neural network r that attentively recognizes text. A coordinate map is constructed to introduce spatial-awareness.

Our main contributions are summarized as follows:

• We present an end-to-end deep neural-based model that utilizes an attention mechanism to recognize both regular and irregular text. The proposed method outperforms previous methods on two irregular-text dataset: SVT-Perspective and CUTE80 with a large margin, while achieves highly-competitive performance on several regular-text datasets.

• We introduce an auxiliary dense character detection task using a FCN to learn text-specific visual patterns, and an alignment loss to encourage an effective attention model. Experiments show that the proposed two components accelerate training and improve performance.

• We develop a large-scale synthetic dataset that contains irregular scene text with character-level bounding box annotations. Hopefully such a dataset will support future studies in this area.

2 Related Work

Scene Text Recognition: Several various methods have been proposed for scene text recognition. Conventional methods often follow a pipeline where individual characters are first detected and recognized and then combined into words based on a set of heuristic rules or a language model. Early work relied on low-level features for character detection and recognition. For example, Neumann et al. [2012] defined a set of handcrafted features such as aspect ratio, hole area ratio, etc. to train a Support Vector Machine classifier. HOG descriptors[Wang *et al.*, 2011; Zhu *et al.*, 2006] were extracted as features to train a character classifier which is then applied to the cropped word image in a sliding-window manner. However, the performance of these methods is limited by the low capability of handcrafted features in terms of expressiveness. With the recent advances in neural-based models, many researchers explored deep neural architectures and achieved better results. In [Bissacco *et al.*, 2013], a fully connected network with 5 hidden layers was employed for character recognition, after which a n-gram approach was used for language modeling. Wang et al. [2012] proposed a CNN to recognize character and a non-maximum suppression method to obtain final word predictions. In [Jaderberg *et al.*, 2014b], a weight-shared CNN with Maxout non-linearity [Goodfellow *et al.*, 2013] was applied for both text/non-text classification and character classification. A word-breakpoints score function was subsequently optimized to obtain word predictions.

The aforementioned pipeline requires the segmentation of individual characters, which can be very challenging because of the complicated background clutter and the inadequate distance between consecutive characters. The performance is therefore limited. To circumvent the need for explicitly isolating characters, several recent work casts scene text recognition problem as a sequential labeling problem, where text is represented by a sequence of characters. [He *et al.*, 2016; Shi *et al.*, 2016a] proposed using RNN for sequential predictions based on visual features learned by a deep CNN. A CTC Loss [Graves *et al.*, 2006] was adopted to calculate the conditional probability between the predicted and the target sequences. Since CTC Loss is only defined for 1-dimensional (1D) sequence, their model is not adequately generalizable to reading irregular text, where characters are arranged on a 2D image plane.

Lee and Osindero [2016] proposed a R^2AM model, where a recursive CNN was operated to learn broader contextual information, then an attention model was applied to perform "soft" 1D feature selection and decoding. Although the attention model has the potential to perform 2D feature selection [Xu *et al.*, 2015], we show in experiments that directly training R^2AM on irregular text is difficult because of the non-horizontal character placement. Our model generalizes R^2AM to performing 2D attentive feature selection with the help of the proposed dense character detection task and the attention alignment loss. Shi et al. [2016b] attempted to recognize irregular text by first rectifying curved or perspectively distorted text to obtain approximately regular text, then recognizing the rectified image. However, with the proposed 2D form attention mechanism in this work, the rectification step becomes unnecessary. Furthermore, we show that the proposed method is capable of recognizing text with more extreme distortions, in which case the rectification module in [Shi *et al.*, 2016b] fails to generate satisfying correction.

Fully Convolutional Networks: Fully convolution networks (FCN) was first proposed by Long et al. [2015] aiming at pixel-wise prediction for semantic segmentation task.

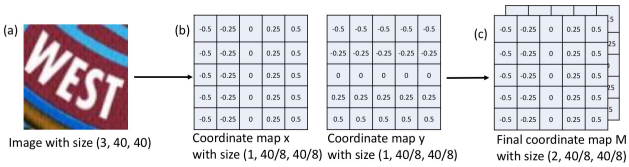

Figure 3: A simple example showing the process of building a coordinate map M. For a 40-by-40 color image, M is a 5-by-5 2-channel image. Each location in M is filled with the normalized coordinates.

Many variations are introduced afterwards that have led to rapid improvement of accuracy [Pinheiro *et al.*, 2016; Noh *et al.*, 2015]. Recently, FCN-based methods have gained much attention in object detection and text detection community as well. For example, Li et al. [2016] used FCN and a position-sensitive Region-of-Interest (RoI) pooling layer to obtain class prediction for each candidate region. In [Gupta *et al.*, 2016], a fully-convolutional regression network was stacked on top a CNN to detect text. We introduce a novel dense character detection task into our end-to-end text recognition framework using FCN. However, the purpose is not to localize individual characters (which could be a very challenging task) as in the conventional pipeline, but rather to help learning better visual representations.

3 Attentive Text Recognition

In this section, we describe the architecture of our model. Overall, the model takes an $W \times H$ image x and outputs a sequence of characters $C = \{C_1, C_2, \cdots, C_T\}$ depicted.

As shown in Figure 2, our model consists of three components: 1) a convolutional feature extractor network f that learns high-level visual representations $f(x)$ from an input image; 2) a deep fully-convolutional neural network g that takes $f(x)$ as input and outputs pixel-wise predictions $\hat{y} = g(f(x))$, where $\hat{y}_{(i,j)} \in \{0, 1\}$ $(1 \leq i \leq W, 1 \leq j \leq H)$ indicates whether the location (i, j) is inside a character's bounding box; 3) a recurrent neural network r that attentively decodes the learned representations and a spatial-aware coordinate map M into a sequence of characters C. During testing, g is omitted in order to save computation costs. The proposed model is end-to-end: it takes an input image and outputs the corresponding word, precluding any pre-processing and post-processing steps. Both the input image and the word depicted can be of varying size or length.

Deep CNNs are good at learning highly semantic features, however, such feature are translation invariant. Spatial information can provide useful guidance to moving attention. For instance, if the attention model is "focusing" on the leftmost character at present, then it is expected to move its attention to the right part at the next time-step. To introduce spatial-awareness, we construct a 2-channel coordinate map M which has the same height h and width w as $f(x)$. Each location in M is filled with the normalized coordinate: $M(u, v) = [(u - w/2)/w, (v - h/2)/h]$. Figure 3 illustrates the construction process via a simple example.

Recognizing text can be essentially considered as a task of modeling sequence interdependencies and learning a mapping between region-of-interests (attention) and characters. Therefore we introduce an attention-based RNN r which is the key component that enables irregular text reading. The decoder r generates one character at a time, hence decomposes the probability of yielding C as:

$$\log \mathrm{P}(C|x) = \sum_{t=1}^{T} \log \mathrm{P}(C_t|C_{<t}, V) \tag{1}$$

where $C_{<t}$ denotes characters before C_t and V is the concatenation of the learned visual representations $f(x)$ and a convolutional transformation (implemented by a single convolution layer with an output channels of 128) of M along the number-of-channel dimension.

The conditional probability can be parameterized as:

$$\mathrm{P}(C_t|C_{<t}, V) = \mathrm{Softmax}(\mathcal{T}(h_t)) \tag{2}$$

where $\mathcal{T}$ is a transformation function (e.g. a feedforward neural network) that outputs a vocabulary-sized vector, and h_t is the hidden state of r. A variation of RNN – Gated Recurrent Unit (GRU) [Cho *et al.*, 2014] is used to model long-term dependencies. As a result, h_t can be computed as:

$$h_t = \mathrm{GRU}(h_{t-1}, C_{t-1}, ctx_t) \tag{3}$$

with ctx_t being the context vector that is a dynamic representation of the relevant part of V at time-step t. We adopt a deterministic 2D soft attention mechanism where ctx_t is the weighted sum of the underneath features:

$$ctx_t = \sum_{i=1}^{w} \sum_{j=1}^{h} \alpha_{(i,j)}^{t} V_{(i,j)} \tag{4}$$

The weight $\alpha_{(i,j)}^{t}$ for each $V_{(i,j)}$ is computed by:

$$\alpha_{(i,j)}^{t} = \frac{\exp(e_{(i,j)}^{t}/\tau)}{\sum_{u=1}^{w}\sum_{v=1}^{h} \exp(e_{(u,v)}^{t}/\tau)} \tag{5}$$

$$e_{(i,j)}^{t} = f_{att}(h_{t-1}, V_{(i,j)}) \tag{6}$$

where τ is a temperature hyper-parameter and $e_{(i,j)}^{t}$ is the alignment score which indicates how relevant the visual representation at $V_{(i,j)}$ is to the decoded character C_t. Low temperature will result in a more concentrated attention α^t. The alignment function f_{att} is parameterized by a single layer multi-layer perceptron such that:

$$f_{att}(h_{t-1}, V_{(i,j)}) = v^{\intercal} \tanh(W h_{t-1} + U V_{(i,j)}) \tag{7}$$

where v, W and U are weight matrices to be learned.

The decoding loss L_{dec} is defined as the negative log likelihood to measure the differences between the predicted and the target character sequence:

$$L_{dec} = -\log \mathrm{P}(C|x) \tag{8}$$

Directly optimizing L_{dec} is difficult due to the model complexity. Therefore an auxiliary dense character detection task is introduced to help learning text-specific visual patterns. We further define an attention alignment loss L_{att} (Section 3.2) to penalize attention when they are not consistent with the location of the corresponding character. During training, the objective is formulated as:

$$L = L_{dec} + \lambda_1 L_{fcn} + \lambda_2 L_{att} \tag{9}$$

where L_{fcn} is the dense character detection loss (Section 3.1). The hyper-parameters λ_1 and λ_2 are meant to balance the three terms.

3.1 FCN for Better Representation Learning

Since the proposed model contains an attention-based RNN on top of a deep CNN, it can be difficult to train. Many researchers in image captioning community (e.g. [Xu *et al.*, 2015]) tackle this problem by using pretrained models to obtain visual representations, hence the major workload is on the training of the RNN. However, based on our experimental results, models trained on large-scale "objects" datasets like ImageNet [Deng *et al.*, 2009] are not optimal for text recognition/detection task. Using the convolutional part of a pretrained VGG16 [Simonyan and Zisserman, 2014] model as our feature extractor f can not lead to model convergence (Figure 5). We hypothesize that models trained on ImageNet emphasize more on semantic objects such as face or body, while text is characterized by the combination of various low-level strokes.

One may consider using models trained for character recognition task as f such as that in [Jaderberg *et al.*, 2014b]. However, since these models aim at classifying isolated characters, training samples are tightly cropped characters. As a consequence, broader contextual information is not fully exploited. Such information is of vital importance. In natural images, consecutive characters may occur very close to each other, therefore a successful text recognition system is expected to distinguish characters with the presence of surrounding characters. Furthermore, for irregular text, surrounding characters are important cues for finding the orientation of the text.

We hereby introduce a dense character detection task using FCN. This task is helpful for utilizing contextual information, as words instead of isolated characters are fed as input. The goal is to obtain a pixel-wise character/non-character prediction $\hat{y} = g(f(x))$ based on the learned visual features $f(x)$. To obtain precise character detection results, $f(x)$ is encouraged to capture text-specific information with the presence of surrounding characters or background noises in its neurons' receptive fields. Since spatial information within a receptive field is largely lost during pooling in f, we adopt unpooling technique [Noh *et al.*, 2015] which reuses the pooled "index" to retain spatial information during upscaling in g.

The dense character detection loss L_{fcn} is a binary softmax loss that is performed at each location. The groundtruth $y_{(i,j)}$ is assigned to 1 if location (i, j) is inside a character's bounding box.

3.2 Attention Alignment Loss

In the early stage of model training, parameters are quite random, leading to an ineffective attention model and therefore incorrect predictions. An intuitive solution is to introduce extra guidance for attention model. For tasks like image captioning or visual question answering, it is difficult to define the groundtruth for attention. Words like "a", "the" or some adjectives can hardly be assigned to a specific relevant region. However, for text recognition task, there is naturally a clear corresponding relationship.

Therefore we can construct the groundtruth for attention $\alpha = \{\alpha_{(i,j)}\}$ in the following way (time-step t is omitted in notations for brevity): Given a character C and its bounding box $b = \{x_b, y_b, w_b, h_b\}$ represented by the center coordinate (x_b, y_b) and the size (w_b, h_b), we assume a truncated Gaussian distribution for $\alpha^{gt} = \{\alpha^{gt}_{(i,j)}\}$:

$$\alpha^{gt}_{(i,j)} = \frac{\alpha^{gt'}_{(i,j)}}{\sum_{i=1}^{w}\sum_{j=1}^{h}\alpha^{gt'}_{(i,j)}} \tag{10}$$

$$\alpha^{gt'}_{(i,j)} = \mathcal{N}\left((i,j)^\intercal|\mu,\Sigma\right) \tag{11}$$

$$\mu = (x_b, y_b)^\intercal \quad \Sigma = \begin{bmatrix} w_b^2/4 & 0 \\ 0 & h_b^2/4 \end{bmatrix} \tag{12}$$

where w and h are width and height of the visual representations $f(x)$.

The proposed model estimates attention α at each time-step during training, which can be seen as another 2D discrete distribution. Hence an attention alignment loss L_{att} can be introduced to measure the disagreement between these two distributions:

$$L_{att} = l(\alpha, \alpha^{gt}) \tag{13}$$

Multiple choices exist for function $l(\cdot,\cdot)$. For instance, one can simply define l as the element-wise $L1$ or $L2$ loss:

$$l(\alpha, \alpha^{gt}) = \frac{1}{wh}\sum_{i=1}^{w}\sum_{j=1}^{h}|\alpha_{(i,j)} - \alpha^{gt}_{(i,j)}| \quad \text{or} \tag{14}$$

$$l(\alpha, \alpha^{gt}) = \frac{1}{2wh}\sum_{i=1}^{w}\sum_{j=1}^{h}(\alpha_{(i,j)} - \alpha^{gt}_{(i,j)})^2 \tag{15}$$

or as a Kullback-Leibler (KL) divergence:

$$l(\alpha, \alpha^{gt}) = D_{KL}(\alpha^{gt}||\alpha) \tag{16}$$

An alternative choice is to use the Wasserstein distance (WD), also known as Earth Mover's distance. Formally, for two distribution P_1 and P_2, the 2nd WD is defined as:

$$WD^2(P_1, P_2) = \inf_{\gamma\in\Gamma(P_1,P_2)} E_{(x,y)\sim\gamma}\,(x-y)^2 \tag{17}$$

where $\Gamma(P_1, P_2)$ denotes the collection of all joint distributions $\gamma(x, y)$ having P_1 and P_2 as marginal. Intuitively, if P_1 and P_2 are viewed as unit amount of dirt piled on their domain respectively, WD indicates the minimum mass that needs to be transported in order to transform one distribution into another. WD possesses many advantageous properties over KL divergence, for instance, WD is insensitive to small oscillations, making the measure more robust to noise [Ni *et al.*, 2009]. When P_1 and P_2 are 1D probability distributions, a closed-form solution exists for WD and its gradient. However, for higher dimensional (e.g. 2D in our case) distributions, the computation of WD and its gradient is demanding. To speedup computation, we follow [Julien *et al.*, 2011] where an efficient Sliced Wasserstein distance (SWD) is considered to approximate WD. For d-dimensional distributions:

$$SWD(P_1, P_2) = \sum_{\theta\in\Omega} WD(P_1^\theta, P_2^\theta) \tag{18}$$

where Ω is the unit sphere in $\mathbb{R}^d$; P_1^θ and P_2^θ are the projected distributions along the direction θ. Equation 18 means that we can approximate WD by summing up a series of 1D Wasserstein distance, which has closed-form solution.

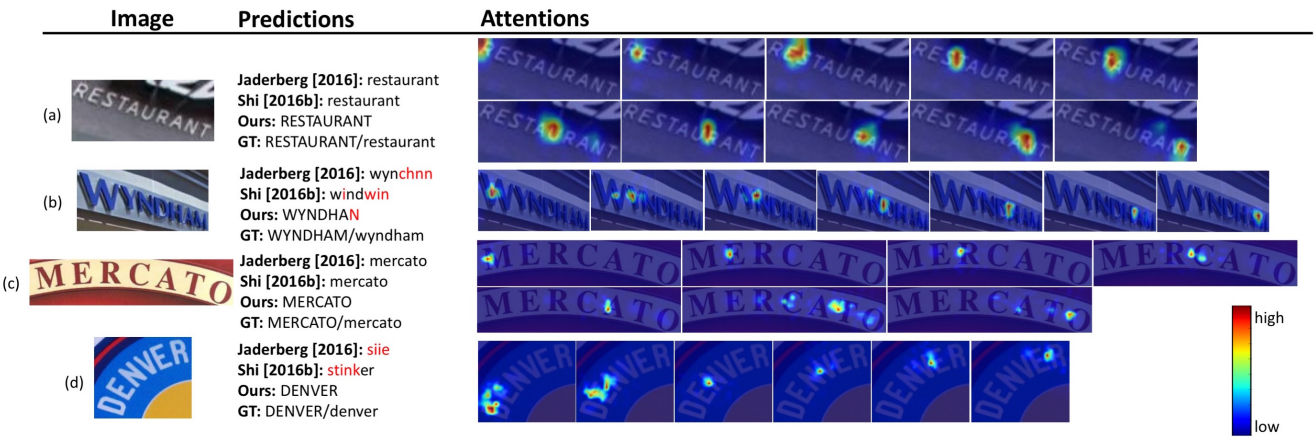

Figure 4: Examples of irregular text recognition. Attention at each time-step are also shown. GT represents groundtruth labels.

3.3 Synthetic Dataset

Jaderberg et al. [2014a] proposed a synthetic data generator to produce regular scene text for training. Following a similar method, we generate a large-scale synthetic dataset containing perspectively distorted and curved text. Different from [Jaderberg *et al.*, 2014a], we also record the character-level bounding boxes to provide guidance for attention. Such dataset will be made public to support future research for irregular text reading.

4 Implementation Details

The architecture of our convolutional feature extractor f is similar to the convolutional part of the VGG16 model. It consists of a sequence of convolutional layers with a kernel size of 3×3, each followed by a Batch Normalization (BN) layer and a Rectify Linear Unit (ReLU) layer. Down-sampling is done by three max-pooling layers each with a pooling size of 2×2. For an input image x of size $W \times H$, $f(x)$ becomes a feature map of size $Ch \times W/8 \times H/8$ where Ch is the number of channels. The FCN g consists of a series of fully-convolutional layers with a kernel size of 3×3, each followed by a BN layer and a ReLU layer. Up-sampling is done by unpooling layers. The hyper parameters λ_1 and λ_2 in our training objective L are set to 10 at the beginning and decrease throughout training. To approximate WD, we project the 2D attention weights along 4 directions: 0° (horizontal), 90° (vertical), 45° and -45°. Beam Search with a window size of 3 is used for decoding in r. The proposed model is trained in an end-to-end manner using stochastic gradient decent. We adopt AdaDelta [Zeiler, 2012] to automatically adjust the learning rate.

5 Experiments

We first conduct ablation experiments to carefully investigate the effectiveness of the model components. After that, we evaluate our model on a number of standard benchmark datasets for scene text recognition, and report word prediction accuracy. Two of these datasets contain images with irregular text, while the rest datasets mostly contain regular text.

5.1 Datasets

This section describes the datasets used in our experiments. Following [Wang *et al.*, 2011], each image may be associated with a lexicon containing a number of candidate words for the purpose of refining the prediction.

SVT-Perspective [Quy Phan *et al.*, 2013] contains 639 cropped images for testing. Images are picked from side-view angle snapshots in Google Street View, therefore one may observe severe perspective distortions.

CUTE80 [Risnumawan *et al.*, 2014] is specifically collected for evaluating the performance of curved text recognition. It contains 288 cropped natural images for testing.

ICDAR03 [Lucas *et al.*, 2003] contains 860 cropped images for testing. For fair comparison, images with non-alphanumeric characters or have less than three characters are discarded, following [Wang *et al.*, 2011].

SVT [Wang *et al.*, 2011] contains 647 cropped images collected from Google Street View. Many images in SVT suffer from low resolution and challenging lighting conditions.

III5K [Mishra *et al.*, 2012] contains 3000 cropped images for testing. Images are collected from the Internet.

5.2 Ablation Experiments on Model Components

Figure 5 (Left) shows the decoding loss L_{dec} when using different approaches to obtain the feature extractor f. If we use the weight of a pretrained VGG16 model to initialize f and keep it fixed, the loss curve will not decrease after reaching a plateau. This justifies our hypothesis that models trained on ImageNet-like data can not capture enough visual cues that are characterize text. Similar results can be observed when using a pretrained Maxout model [Jaderberg *et al.*, 2014b] which is originally proposed for isolated character recognition. Although this time the model reaches a smaller loss, it still gets stuck at a later stage. If we train the model from scratch, the loss will decrease. Such phenomenon suggests that the irregular text reading task requires visual representations that are very different from those learned through isolated character recognition. Finally, incorporating a dense character detection task using FCN in scratch training leads to a notable speedup and a lower decoding loss.

Figure 5 (Right) shows the decoding loss L_{dec} when different kinds of attention alignment loss L_{att} is applied. As

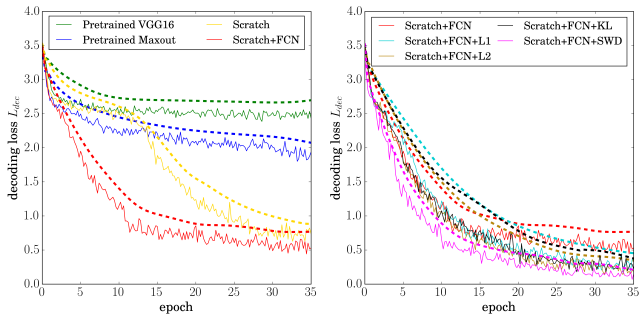

Figure 5: Decoding loss L_{dec} on training (solid lines) and validation (dashed lines) set. **Left**: Loss curves when using different approaches to obtain the feature extractor f (Pretrained VGG16, Pretrained Maxout, Scratch and Scratch+FCN"). **Right**: Loss curves when using different attention alignment loss (None, $L1$, $L2$, KL divergence or SWD).

Method	SVT-Perspective			CUTE80
	50	Full	None	
[Wang *et al.*, 2011]	40.5	26.1	-	-
[Mishra *et al.*, 2012]	45.7	24.7	-	-
[Wang *et al.*, 2012]	40.2	32.4	-	-
[Quy Phan *et al.*, 2013]	75.6	67.0	-	-
[Shi *et al.*, 2016a]	92.6	72.6	66.8	54.9
[Jaderberg *et al.*, 2016]	-	-	-	42.7
[Shi *et al.*, 2016b]	91.2	77.4	71.8	59.2
Ours	**93.0**	**80.2**	**75.8**	**69.3**

Table 1: Recognition accuracies on two irregular text datasets: SVT-Perspective and CUTE80. "50" and "Full" represent the size of lexicons ("Full" means that we use all words in the dataset), while "None" represents recognition without using a lexicon.

we can see, adding L_{att} results in a lower decoding loss. We argue that introducing supervision directly on attention leads to a more effective attention model, which is of vital importance for reading irregular text. We further compare the effects of using different types of L_{att}. As shown in Figure 5 (Right), SWD yields the lowest decoding loss on both training and validation set. The physical implications of WD makes it very suitable for the task of regressing the estimated attention to the groundtruth attention. The difference among using $L1$, $L2$ or KL loss is marginal.

The "Scratch+FCN+SWD" model in Figure 5 (Right) is selected to report recognition results on benchmark datasets.

5.3 Results on Irregular Text

Table 1 summarized the recognition accuracies on two irregular text datasets: SVT-Perspective and CUTE80. On SVT-Perspective dataset, the proposed model achieves the highest accuracies. We observe that a large portion of test images in SVT-Perspective dataset have a small amount of perspective distortion, therefore models that are only trained on regular text (e.g. [Shi *et al.*, 2016a]) can also achieve competitive results. However, on CUTE80 dataset where many images contain curved text, the proposed model outperforms previous methods with a large margin. The irregular character placement and the rotated characters pose a challenge to regular-text recognition methods.

In Figure 4, we show several examples illustrating the movement of attention when recognizing irregular text. The proposed model successfully focuses on the correct character at each time-step, even on some challenging images with significant perspective distortion (Figure 4(b)) or large curve angle (Figure 4(d)). In these cases, the rectification module in [Shi *et al.*, 2016b] fails to produce a satisfying correction.

Method	IC03-50	SVT-50	III5K-50	III5K-1k
abbyy [Wang *et al.*, 2011]	56.0	35.0	24.3	-
[Wang *et al.*, 2011]	76.0	57.0	64.1	57.5
[Yao *et al.*, 2014]	88.5	75.9	80.2	69.3
[Wang *et al.*, 2012]	90.0	0.0	-	-
[Jaderberg *et al.*, 2014b]	96.2	86.1	-	-
[Jaderberg *et al.*, 2014a]	**98.7**	95.4	97.1	92.7
[He *et al.*, 2016]	97.0	93.5	94.0	91.5
[Shi *et al.*, 2016a]	**98.7**	**96.4**	97.6	94.4
[Shi *et al.*, 2016b]	98.3	95.5	96.2	93.8
Ours	97.7	95.2	**97.8**	**96.1**

Table 2: Recognition accuracies on several regular text datasets. "50" and "1K" represent the size of lexicons.

5.4 Results on Regular Text

In Table 2, we compare the proposed model with other methods for regular text recognition. Our model achieves the best results on III5K-50 and III5K-1K datasets. III5K contains more test images, many of which suffer from perspective distortion and curving effects. Our model falls behind [Jaderberg *et al.*, 2014a] slightly on IC03-50 dataset. However, Jaderberg et al. [2014a] casts text recognition as an image classification problem where each word is defined as a class label. Consequently, their model can not recognize out-of-vocabulary words. Shi et al. [2016a] outperforms our model on IC03-50 and SVT-50 datasets. However, they treat input images as a 1D horizontal sequence of visual features, therefore irregular text with large rotation or curve angles can not be successfully recognized. On the contrary, the proposed model is capable of reading both regular and irregular text.

6 Conclusions

We present an end-to-end model which attentively reads both regular and irregular text. Recognizing irregular text in natural scene is addressed by first learning text-specific visual representations, then decoding the learned representations into a character sequence via an attention-based RNN. To facilitate training, we propose 1) a dense character detection task using a FCN for representation learning and 2) an alignment loss to provide guidance for attention. These two components prove crucial for achieving fast model convergence and high performance. We also visualize the attention weights to better analyze the model's behavior. Future directions would be to combine the proposed text recognition model with a text detection method for a full end-to-end system.

Acknowledgments

We gratefully acknowledge partial support from NSF grant CCF 1317560 and a hardware grant from NVIDIA.

References

[Bissacco *et al.*, 2013] Alessandro Bissacco, Mark Cummins, Yuval Netzer, and Hartmut Neven. Photoocr: Reading text in uncontrolled conditions. In *ICCV*, 2013.

[Cho *et al.*, 2014] Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*, 2014.

[Deng *et al.*, 2009] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

[Goodfellow *et al.*, 2013] Ian Goodfellow, David Warde-farley, Mehdi Mirza, Aaron Courville, and Yoshua Bengio. Maxout networks. In *ICML*, 2013.

[Graves *et al.*, 2006] Alex Graves, Santiago Fernández, Faustino J Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *ICML*, 2006.

[Gupta *et al.*, 2016] Ankush Gupta, Andrea Vedaldi, and Andrew Zisserman. Synthetic data for text localisation in natural images. In *CVPR*, 2016.

[He *et al.*, 2016] Pan He, Weilin Huang, Yu Qiao, Chen Change Loy, and Xiaoou Tang. Reading scene text in deep convolutional sequences. In *AAAI*, 2016.

[Jaderberg *et al.*, 2014a] Max Jaderberg, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Synthetic data and artificial neural networks for natural scene text recognition. In *NIPS Workshop*, 2014.

[Jaderberg *et al.*, 2014b] Max Jaderberg, Andrea Vedaldi, and Andrew Zisserman. Deep features for text spotting. In *ECCV*. Springer, 2014.

[Jaderberg *et al.*, 2016] Max Jaderberg, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Reading text in the wild with convolutional neural networks. *IJCV*, 116, 2016.

[Julien *et al.*, 2011] Rabin Julien, Gabriel Peyré, Julie Delon, and Bernot Marc. Wasserstein barycenter and its application to texture mixing. In *SSVM*. Springer, 2011.

[Lee and Osindero, 2016] Chen-Yu Lee and Simon Osindero. Recursive recurrent nets with attention modeling for ocr in the wild. In *CVPR*. IEEE, 2016.

[Li *et al.*, 2016] Yi Li, Kaiming He, Jian Sun, et al. R-fcn: Object detection via region-based fully convolutional networks. In *NIPS*, 2016.

[Long *et al.*, 2015] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015.

[Lucas *et al.*, 2003] Simon M Lucas, Alex Panaretos, Luis Sosa, Anthony Tang, Shirley Wong, and Robert Young. Icdar 2003 robust reading competitions. In *ICDAR*, 2003.

[Mishra *et al.*, 2012] Anand Mishra, Karteek Alahari, and CV Jawahar. Scene text recognition using higher order language priors. In *BMVC*, 2012.

[Neumann and Matas, 2012] Lukáš Neumann and Jiří Matas. Real-time scene text localization and recognition. In *CVPR*. IEEE, 2012.

[Ni *et al.*, 2009] Kangyu Ni, Xavier Bresson, Tony Chan, and Selim Esedoglu. Local histogram based segmentation using the wasserstein distance. *IJCV*, 84, 2009.

[Noh *et al.*, 2015] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. In *ICCV*. IEEE, 2015.

[Pinheiro *et al.*, 2016] Pedro O Pinheiro, Tsung-Yi Lin, Ronan Collobert, and Piotr Dollár. Learning to refine object segments. In *ECCV*. Springer, 2016.

[Quy Phan *et al.*, 2013] Trung Quy Phan, Palaiahnakote Shivakumara, Shangxuan Tian, and Chew Lim Tan. Recognizing text with perspective distortion in natural scenes. In *ICCV*, 2013.

[Risnumawan *et al.*, 2014] Anhar Risnumawan, Palaiahankote Shivakumara, Chee Seng Chan, and Chew Lim Tan. A robust arbitrary text detection system for natural scene images. *Expert Systems with Applications*, 41, 2014.

[Shi *et al.*, 2016a] Baoguang Shi, Xiang Bai, and Cong Yao. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *CVPR*, 2016.

[Shi *et al.*, 2016b] Baoguang Shi, Xinggang Wang, Pengyuan Lyu, Cong Yao, and Xiang Bai. Robust scene text recognition with automatic rectification. In *CVPR*. IEEE, 2016.

[Simonyan and Zisserman, 2014] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[Wang *et al.*, 2011] Kai Wang, Boris Babenko, and Serge Belongie. End-to-end scene text recognition. In *ICCV*. IEEE, 2011.

[Wang *et al.*, 2012] Tao Wang, David J Wu, Adam Coates, and Andrew Y Ng. End-to-end text recognition with convolutional neural networks. In *ICPR*. IEEE, 2012.

[Xu *et al.*, 2015] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron C Courville, Ruslan Salakhutdinov, Richard S Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*, 2015.

[Yao *et al.*, 2014] Cong Yao, Xiang Bai, Baoguang Shi, and Wenyu Liu. Strokelets: A learned multi-scale representation for scene text recognition. In *CVPR*, 2014.

[Zeiler, 2012] Matthew D Zeiler. Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*, 2012.

[Zhu *et al.*, 2006] Qiang Zhu, Shai Avidan, Meichen Yeh, and Tim Kwang-Ting Cheng. Fast human detection using a cascade of histograms of oriented gradients. In *CVPR*, 2006.