

**MRZ code extraction from visa and passport documents using convolutional neural network**

This is an preprint of <https://doi.org/10.1007/s10032-021-00384-2>

Yichuan Liu\*, Hailey James, Otkrist Gupta and Dan Raviv

Lendbuzz

[yichuan.liu@lendbuzz.com](mailto:yichuan.liu@lendbuzz.com)

125 High Street, Boston, Massachusetts

\*corresponding author

# MRZ code extraction from visa and passport documents using convolutional neural networks

Yichuan Liu · Hailey James · Otkrist Gupta · Dan Raviv

Received: date / Accepted: date

**Abstract** Detecting and extracting information from the Machine-Readable Zone (MRZ) on passports and visas is becoming increasingly important for verifying document authenticity. However, computer vision methods for performing similar tasks, such as optical character recognition (OCR), fail to extract the MRZ from digital images of passports with reasonable accuracy. We present a specially designed model based on convolutional neural networks that is able to successfully extract MRZ information from digital images of passports of arbitrary orientation and size. Our model achieves 100% MRZ detection rate and 99.25% character recognition macro-f1 score on a passport and visa dataset.

**Keywords** First keyword · Second keyword · More

## 1 Introduction

In domains such as finance, immigration and administration, digital copies of passports are playing an increasingly important role in identity and information verification and fraud detection. However, automatic information retrieval from passports and visas can be difficult due to non-uniform passport and visa layouts. Information such as name, birth date, expiration date, and issue date appear in a variety of formats and locations on passports and visas from different issuing authorities. Additionally, unlike physical passports and visas which can be examined for authenticity, digital copies of these documents present a lower barrier to

forgery and manipulation. Simple image editing software can be used to alter key details on the passport or visa for purposes of fraud.

The Machine-Readable Zone (MRZ) on passports and visas is critical for combating both of these challenges. For purposes of information verification, the MRZ presents key information in a pre-specified format and location. As a fraud example, some of our customers uploaded the passport image they found on the internet instead of their own documents. Being able to read the MRZ and compare it with the information entered by our customers thus serves as an important step in fraud detection. Similarly, the format of the MRZ makes it more difficult to manipulate than the rest of the passport, requiring domain knowledge and greater attention to detail. Locating and extracting passport and visa MRZ thus presents an important and unique application for computer vision.

We propose a novel neural network model designed specifically for handling MRZ text, with characteristics designed to overcome the challenges unique to MRZ extraction. Specifically, we design an end-to-end trainable MRZ detector and extractor using MobileNetV2 as the backbone and added atrous spatial pyramid pooling layers to enhance receptive fields. For better handling of passport images of various sizes, we propose a novel system in which the first "coarse" model extracts the MRZ bounding box and the second "fine" model refines bounding box prediction and extracts the MRZ text. This system design offers the additional benefit of decreasing the memory and time required for detection. Our proposed system results in 100% MRZ detection rate and 99.25% character recognition macro-f1 score on digital images of passports and visas.

---

Yichuan Liu  
125 High Street, Boston, Massachusetts  
E-mail: yichuan.liu@lendbuzz.com

Dan Raviv  
E-mail: dan.raviv@lendbuzz.com

## 2 Background

### 2.1 Machine-Readable Zone (MRZ)

The Machine-Readable Zone (MRZ) appears on passports and visas of most countries to facilitate robust data extraction and processing. Because passports from different states vary in script, style, and format, the MRZ provides a simple way to extract key details from the passport, including the name, passport number, nationality, date of birth, sex, and passport expiration date. Generally appearing near the beginning of the passport on the identity page, the MRZ text typically appears as two 44-character lines on the bottom of the page. While alternative MRZ formats are employed in documents such as ID card and visa issued by some countries, we limit this work to consider only this MRZ format most commonly found in passport and US visa images as typically uploaded by our customers. The MRZ consists only of the Arabic numerals (digits 0-9), the capital letters of the Latin alphabet ('A', 'B', 'C', ...), and the filler character '<'. While historically used for quickly extracting the important information from a variety of passports, the MRZ is becoming useful for document verification and manipulation detection. For example, businesses and states can verify that the information encoded in the MRZ matches the information in the visual zone (VZ) of the passport. While highly-motivated and skilled forgers can additionally alter the MRZ, validation of the MRZ information is a simple, low-cost method for detecting basic manipulations, such as name, expiration, or birth date changes. As photographs of passports gain popularity as a method for verifying identity, accurately and quickly extracting the passport MRZ becomes an essential part of the identity verification pipeline.

## 3 Related Work

Models leveraging advances in deep learning, such as convolutional neural networks (CNNs), have been successfully employed in similar tasks, such as determining the region of interest (ROI) of a photograph [34] and optical character recognition (OCR) [52]. Among these, MRZ extraction from digital passport images is most related to work in detecting and extracting text in natural scenes.

### 3.1 Text Detection in Natural Scenes

Several techniques in computer vision have been developed or leveraged for improved performance in text

scene detection. [33] and [58] used image binarization to segment text regions. [39] [10] and [14] used Maximally Stable Extremal Region (MSER) to improve detection. [12] and [40] used morphological operations to segment text regions. [41] and [43] used Histogram of Oriented Gradients (HOG) for improved performance. [23] and [42] used color properties to both detect and extract text regions.

More recently, the ICDAR 2015 Robust Reading Competition dataset has provided a valuable benchmark for scene text detection and extraction [22]. Many recent works demonstrated impressive performance on this dataset. [56] use Fully Convolutional Network (FCN) models, trained separately to predict the saliency map of the text regions and the centroid of each character. [48] similarly trained a FCN, but on the segments and links of the text which are combined for final detection. [18] proposed a single shot attention-based mechanism that attempts a coarse to fine approach to text detection. [21] leveraged a weakly supervised framework that uses word annotations to train the character detector. [57] proposed EAST (An Efficient and Accurate Scene Text Detector), which skips intermediate steps like candidate aggregation and word partitioning to directly predict words and text lines. [35] attempted to consider more free-form text examples such as curved text using a FCN to estimate geometric attributes of the scene regions. [27] iterated on the object detector method proposed in [29]. [7] proposed a novel method for scene text detection using instance segmentation. [38] and [9] leveraged the corner points of the text bounding boxes for better segmentation and detection. [53] generated different kernel scales for each text instance in order to split close text instances. [6] combined multi-level features during feature extraction for improved performance. [36] improved performance with an architecture inspired by Mask R-CNN [16]. [28] proposed a module to perform binarization in a segmentation network. [30] trained a network for simultaneous detection and recognition by sharing convolutional features between the two processes. [55] leveraged multiple branches to achieve geometry normalization. [32] built on previous work and incorporate a method to discretize the potential quadrilaterals into various horizontal and vertical positions. So far, [54] achieved the most impressive performance by training on synthetic data, using characters as the basic element, and eliminating ROI pooling.

### 3.2 Passport MRZ Detection and Extraction

While Optical Character Recognition (OCR) based-methods may extract text with reasonably good accuracy, state of the art methods struggle to accurately extract MRZ



Fig. 1: Example passport images. (a) a typical page of the passport contains the 2-line MRZ zone (bottom). The passport page can either occupy only a small part of the image (b) or span the whole image (c).

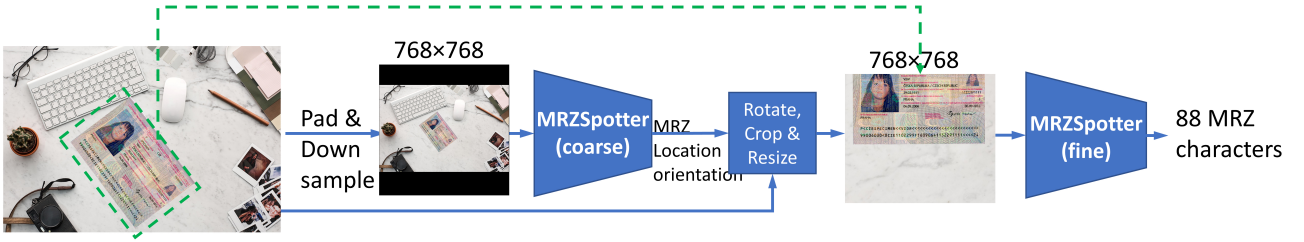


Fig. 2: Overall structure of MRZNet. MRZSpotter (coarse) roughly locates the MRZ region from a down-sampled image whereas MRZSpotter (fine) refines the localization on the original high resolution image and recognizes the MRZ text.

text. This is evidenced by the relatively poor MRZ detection rate of PassportEye [51] which is based on Tesseract OCR [50]. Similarly, models designed for scene text extraction are not naturally well-suited for MRZ extraction. For example, end-to-end scene text recognition models such as FOTS [31] and Mask Textspotter [37] may be able to detect and recognize the MRZ. However, these models are designed to handle text lines with arbitrary number of characters and employed techniques such as LSTM [20] to recognize text. Since common MRZ text found in passport and visa is always 2 lines, 44 characters per line, a specifically designed neural network architecture will likely improve performance. Additionally, typical passport images used for identity verification purposes are taken with a smartphone, resulting in a high resolution images in which the passport appears in various places and at various sizes (see Figure 1), presenting an additional challenge.

In 2011, [3] presented a hardware-based method for portable passport readers for detecting and reading the MRZ of physical passports. [26] proposed a method for extracting the passport MRZ using template matching, but only for images in which the passport is surrounded by a black border. [5] explored optical font recognition for forgery detection in passport MRZs. [44] dis-

cussed methods for correcting or post-processing passport MRZ recognition results. [15] presented an algorithm for reading MRZ images on mobile devices, achieving an MRZ detection rate of 88.18% with 5 frames and 56.1% with single frame, along with a character reading rate of 98.58%. In comparison our model boasts a 100% single frame MRZ detection rate and 99.25% character recognition macro-f1 score on passport and visa images.

## 4 Methodology

MRZNet is a framework that detects and recognizes the MRZ text in images of passports and visas given arbitrary orientation and sizes. This section describes the details of the architecture of MRZNet.

### 4.1 Overall architecture

The overall architecture of MRZNet is illustrated in Figure 2. It includes two sub neural networks, MRZSpotter (coarse) and MRZSpotter (fine), which share similar architectures. The high resolution original image is first padded to be a square and then down-sampled to 768 x 768 as input to MRZSpotter (coarse). MRZSpotter (coarse) localizes the MRZ region and outputs the

bounding box location and orientation. We then rotate the original image to make it upright, crop the image centered at the bounding box center and pad/resize the image accordingly to obtain a  $768 \times 768$  image in which the MRZ region is roughly placed in the center and spans the whole image. This image is then fed into MRZSpotter (fine) for finer localization and MRZ code recognition. We adopt this architecture for handling passport/visa images of arbitrary orientation and sizes. A real world passport/visa image, whether it is scanned or taken from a smart phone, is usually of high resolution. Depending on how the user captures the image, the MRZ region can either occupy only a small region of the image (Figure 1(b)) or it may span the whole image (Figure 1(c)). Feeding the high resolution image directly into a neural network is not only time and memory consuming but may result in poor MRZ code recognition results for images like Figure 1(c). Specifically, localizing these high resolution images would demand a very large receptive field within the neural network, increasing time and memory requirements. On the other hand, feeding a down-sampled image to a neural network would result in poor MRZ recognition results for images like Figure 1(b) because the text will be unrecognizable in low resolution. To solve this dilemma, we propose an architecture that first roughly localizes the MRZ region using a down-sampled image, standardizes the images (see Figure 1(c)) and finally performs MRZ text recognition.

#### 4.2 MRZSpotter

The architecture of MRZSpotter is shown in Figure 3. Because we use CPU to run models in production, we adopt MobileNetV2 [46] as the backbone to reduce computational cost. Similar to EAST, we concatenate up-sampled high-level semantic feature maps with low-level feature maps and merge them gradually in a U-shaped architecture. This way the neural network utilizes the features from different levels and will be able to detect MRZ regions of different sizes. In some examples, the line of text will span the whole image (see Figure 1(c)). For better handling of these images, a larger receptive field is required to look at the "big picture" of the image in order to accurately detect the large text bounding box. We applied atrous spatial pyramid pooling (ASPP) at the end of the MobileNetV2 feature extractor to accommodate these larger receptive fields. ASPP have been previously adopted by [47], [13], [49] and [4] for field-of-view enlargement. To further increase the field-of-view, we stacked multiple layers of ASPP as demonstrated in ResNet [17]. After feature-merging,  $1 \times 1$  convolutional layers are applied to the output to

determine the likelihood that an MRZ region is present in the pixel (the score map), the location of MRZ text boxes (4 channels, distance of the pixel locations to the top, right, bottom and left boundaries of the rectangle, respectively) and the MRZ box rotation angle. The non-maximum suppression algorithm is applied to select the most probable MRZ bounding box. Finally, a recognition branch is applied to the MRZ bounding box and the output map of the feature-merging branch to extract the MRZ text.

#### 4.3 MRZSpotter pipeline

We first extract feature maps from a passport/visa image using a MobileNetV2 backbone. At the end of the stage 4 convolutional layers, MobileNetV2 produces 320 feature maps of size  $24 \times 24$ . We then add four convolutional layers that run in parallel to form a ASPP layer. These four convolutional layers have a dilation rate [4] of 1, 2, 4 and 8. We concatenate the feature maps produced by these four layers (the concatenation layer) and then applied a  $1 \times 1$  convolutional layer to reduce the number of feature maps to 320 before feeding the resulting feature maps to the next ASPP layer. Shortcuts were added between the concatenation layer of two ASPP layers similar to ResNet. After N ASPP layers, we bilinearly up-sample (un-pool) the feature maps to size  $48 \times 48$  before concatenate them with the feature map outputs from the end of stage 3 convolutional layer of MobileNetV2. A  $1 \times 1$  followed by a  $3 \times 3$  convolutional layer is used to fuse these feature maps. We then bilinearly up-sample the resulting feature maps to  $96 \times 96$  sizes and concatenate them with output of the stage 2 convolutional layers of MobileNetV2. After fusing the feature maps with 2 convolutional layers, we bilinearly upsampled them to  $192 \times 192$  sizes and concatenate them with the output of stage 1 convolutional layers of MobileNet V2. Three convolutional layers are then applied to fuse and extract features from these feature maps to produce the output of feature-merging branch, which is composed of 64 feature maps of size  $192 \times 192$ . Similar to EAST[57], for each pixel in the output of feature-merging branch, we apply  $1 \times 1$  convolutional layers at the output branch to produce a 0-1 probability score which indicates the presence of MRZ (the score map) at the pixel, the distance from the top, bottom, left and right of the mrz bounding box to the pixel (MRZ text box map) and the rotation angle of the bounding box (mrz rotation angle map). Because we have  $192 \times 192$  pixels, a total of  $192 \times 192 = 36864$  bounding boxes are produced as a result. We reject those bounding boxes that have a

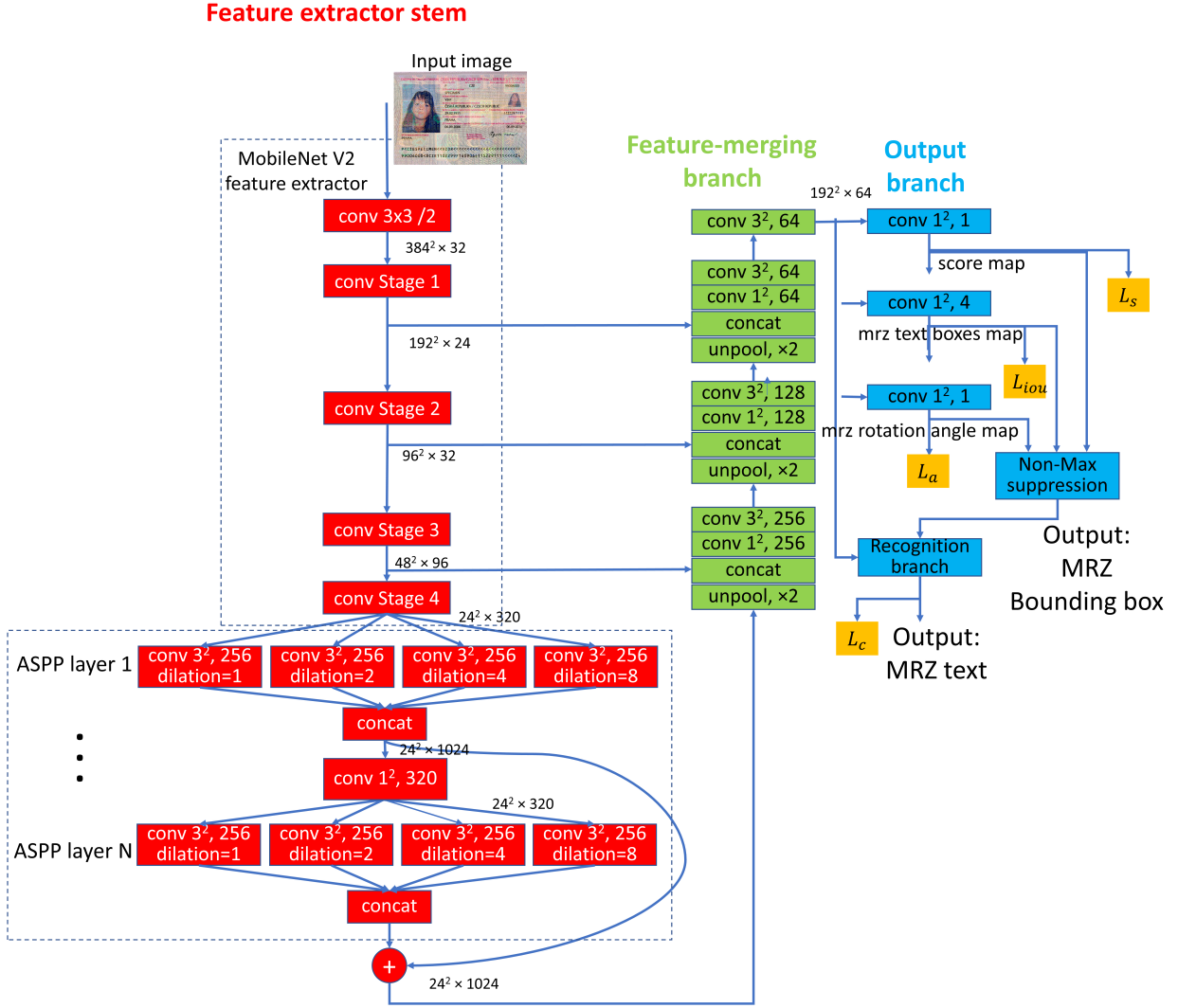


Fig. 3: MRZSpotter with  $N$  atrous spatial pyramid pooling (ASPP) layers. Both MRZSpotter (coarse) and MRZSpotter (fine) used the same architecture and loss as shown in this figure, though with different parameter  $N$ . We stacked  $N$  ASPP layers on top of the last convolutional stage of MobileNetV2 to increase the receptive field and add a text recognition branch in addition to text localization branch.

probability score lower than 0.5 and use non-max suppression (NMS) to fuse the rest of the bounding boxes. The bound box that has the highest score is then selected as input to the recognition branch.

#### 4.4 Recognition branch

Both the MRZSpotter (coarse) and MRZSpotter (fine) include a recognition branch for recognizing MRZ text. Our recognition branch is inspired from [19]. Figure 4 shows the architecture. Given the quadrilateral MRZ region from NMS, we sample a 16 by 352 grid from the convolutional map at the output of feature-merging

branch. Similarly to [19], we used bilinear sampling. More specifically, the feature vector  $v_p$  of a sampling point  $p$  at spatial location  $(p_x, p_y)$ , is calculated as follows:

$$v_p = \sum_{i=0}^3 v_{pi} g(p_x, p_{ix}) g(p_y, p_{iy}) \quad (1)$$

where  $v_{pi}$  refer to the surrounding four points of point  $p$  and  $g(p_1, p_2)$  refers to the bilinear interpolation function.

After extracting the sampling grid, three layers of 3x3 convolution and 2x2 max-pooling are applied to down sample the extracted feature map from 16x352 (points) to 2x44 (lines by characters per line). We dou-

bled the number of channels with each down-sampling. Finally, a 1x1 convolutional layer is applied to reduce the number of channels to 37 (the number of valid characters in MRZ code) and softmax is applied to obtain the probability of occurrence for each of the 88 characters.

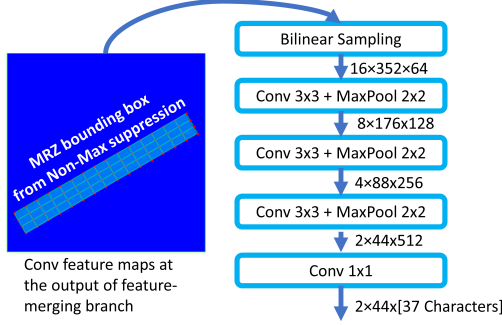


Fig. 4: Architecture of the recognition branch. Instead of adopting LSTM for recognizing arbitrary length text as in typical scene text recognition networks, we bilinearly sample a 16 by 352 grid from the output convolutional layer of EAST and pass it through several full convolutional layers and max pooling layers to reduce the feature map size to 2 by 44 for recognizing the 2 lines, 44 characters per line MRZ text.

#### 4.5 Loss functions

MRZSpotter (coarse) and MRZSPotter (fine) are trained separately with the same loss function:

$$L = L_s + \lambda_g L_g + \lambda_c L_c \quad (2)$$

where  $L_s$  is the loss for score map,  $L_g$  is the loss for geometry,  $L_c$  is the loss for character classification. In our experiment, we set  $\lambda_g$  and  $\lambda_c$  to be 1. For the loss of score map, we used dice loss instead of the balanced cross-entropy loss as adopted by EAST [57] due to its higher performance as reported in [45]:

$$L_s = 1 - \frac{2 \sum_x s_x s_x^*}{\sum_x s_x + \sum_x s_x^*} \quad (3)$$

where  $s_x$  and  $s_x^*$  are predicted score and ground truth score, respectively. For the geometry loss, we adopt the intersection over union (IoU) loss and rotation angle loss as in EAST [57]:

$$L_g = L_{iou} + \lambda_a L_a \quad (4)$$

$$L_{iou} = \frac{1}{\Omega} \sum_{x \in \Omega} IoU(R_x, R_x^*) \quad (5)$$

$$L_a = (1 - \cos(\theta_x, \theta_x^*)) \quad (6)$$

where  $R_x$ ,  $R_x^*$ ,  $\theta_x$  and  $\theta_x^*$  are predicted bounding box, ground truth bounding box, predicted orientation and ground truth orientation, respectively.  $IoU$  is calculated as follows:

$$IoU(R_x, R_x^*) = \frac{R_x \cap R_x^*}{R_x \cup R_x^*} \quad (7)$$

In our experiment, the weight  $\lambda_a$  is set to 10. For the character classification loss, we used the cross-entropy loss:

$$L_c = \sum_{i=0}^c y_i \log(f_i(x)) \quad (8)$$

where  $c$  is the number of possible different characters,  $f_i(x)$  is the network output of class  $i$  for image sample  $x$ ,  $y_i$  is the one hot ground truth label.

#### 4.6 Implementation details

For the MobileNetV2 backbone, we loaded weights pre-trained on the ImageNet dataset [8] before fine-tuning on our MRZ dataset. For training MRZSpotter (coarse), we augmented the dataset by randomly rotating the images in the range of  $[-180^\circ, 180^\circ]$  and randomly padding with black borders so that the new image height is in the range of 1-2 times the height of the original image. We additionally applied random cropping image with the constraint of keeping the MRZ region intact. For training MRZSpotter (fine), we augmented the dataset by randomly rotating the image in the range of  $[-20^\circ, 20^\circ]$  with respect to the upright position. We found that it is important to make sure that the rotation angle is small for MRZSpotter (fine). We then cropped the images so that the cropped region is a square and the MRZ region is roughly centered within image with the left and right borders randomly selected to be 0.05-0.25 times the width of the MRZ region. For both MRZSpotter (coarse) and MRZSpotter (fine), we trained the model for 120 epochs with Adam optimizer [24] and a initial learning rate of 0.0001,  $\beta_1 = 0.9$  and  $\beta_2 = 0.999$ . The learning rate was decreased by a factor of 10 at 60 epochs. The batch size was selected to be 6. The models were trained with a single GeForce RTX 2070 graphic card. The time to train both MRZSpotter (coarse) and MRZSpotter (fine) are approximately 1 day, making it a total of 2 days to train the entire model. Taken together MRZSpotter (fine) and MRZSpotter (coarse), MRZNet has 28.9M parameters. As a comparison, similar deep learning approaches CharNet [54] and FOTS [31] has 89.2M and 35.0M parameters, respectively.

## 5 Experimental evaluation

In this section, we evaluate the performance of MRZNet. We also report results from ablation studies to explore the impact of our design choices.

### 5.1 Dataset

We evaluated our algorithm on a dataset consisting of 4820 passport/visa images. The dataset includes 2687 passport images and 2133 visa images from 85 issuing countries. The visa images are all US issued visa the MRZ of which is 2 lines, 44 characters per line. Table 1 summarizes the data distribution over different countries. The dataset contain real world images that were either scanned by a scanner or taken using a smartphone camera and uploaded to our database by our customers. It may contain perspective distortions, scaling, illumination and resolution variation, even motion blur. It reflects the real world images we encounter on a daily basis. Each of these images contains a single passport or visa. We manually annotated the ground truth bounding box and MRZ text for all images using VGG image annotator [11]. It is common that text documents other than passport and visa appears in an image so training with this dataset allow our model to ignore none MRZ text regions. We used 3482 for training, 723 for validation and 615 for testing. We also tested our approach on the publicly available MRZ dataset MIDV-500 [2] and syntheticMRZ [15]. For MIDV-500, we used the passport images that contain a MRZ zone. We removed images in which MRZ zone is not entirely intact. This resulted in 3335 test images. For syntheticMRZ, we randomly selected 17113 images. For both MIDV-500 and syntheticMRZ, we only include images whose MRZ zone is in the most common format, containing two lines of text, 44 characters each. The file paths to the images for the MIDV-500 and syntheticMRZ dataset is available upon request.

### 5.2 Comparison with existing solution

We compared our MRZNet against existing MRZ recognition solutions: 1) PassportEye [51] which is based on Tesseract OCR [50], 2) MRZ-Detection [25] and 3) UltimateMRZ [1], a deep learning based commercial solution that is based on LSTM [20]. We also compared it with end-to-end neural network based text spotting approaches MaskTextSpotter [37], the EAST based TextSpotter [19] and CharNet [54]. For these approaches, we downloaded code and weights trained on ICDAR2015 from official implementations. It can

Table 1: Distribution of our MRZ dataset across issuing countries

Country	Num. of samples
United States	2194
Brazil	1114
China	931
India	48
Guatemala	45
Venezuela	42
Colombia	33
El Salvador	31
Mexico	31
United Kingdom	29
Ecuador	20
Australia	17
Italy	15
Other	270

be seen from Table 2, Table 3 and Table 4 that our MRZNet outperforms each of the three comparison MRZ detection models as well as other deep learning based end-to-end text retrieval models by a large margin. In Table 2, Table 3 and Table 4, for MRZNet, PassportEye, MRZ-Detection, and UltimateMRZ, the detection rate is defined as the ratio of images whose MRZ character recognition accuracy is higher than 50%, as the ground truth bounding box of SyntheticMRZ dataset is not available, and because PassportEye and MRZ-detection do not output the predicted bounding box. For the three deep learning based approaches, it is rare that the MRZ character recognition accuracy is higher than 50% for an image, so we consider the detection is a success if a text box is found in the MRZ region based on the ground truth bounding box. Figure 5 shows examples of text detection results by the various deep learning based approaches. In addition to these approaches, Hartl et al. [15] achieved character recognition rate of 98.6% on the SyntheticMRZ dataset. Their MRZ detection rate, however, is only 56.1% (single frame) and 88.2% rate (5 frames) whereas our single frame MRZ detection rate is 88.66% on the SyntheticMRZ dataset. One possible explanation of this large performance gap is that most existing algorithms rely on traditional image processing techniques or the output of a general OCR, while our method employs convolutional neural network as feature extractor for end-to-end detection and recognition. Additionally, MRZNet is specifically designed to handle MRZ detection and recognition which assumes a fixed target of two lines of 44 characters each whereas the end-to-end scene text detectors proposed in the literature are designed for text lines of arbitrary length. We also reported the run time of all approaches in Table 5



Fig. 5: MRZ recognition results by end-to-end deep learning approaches. From top to bottom rows: TextSpotter, MaskTextSpotter, CharNet and MRZNet

Table 2: MRZ detection and character recognition (in macro-f1 score) results on our test set for MRZNet and other solutions

Method	MRZ Detection	Character Recog.
PassportEye	26.50%	84.47%
MRZ-Detection	52.36%	95.01%
UltimateMRZ <sup>a</sup>	68.78%	83.04%
TextSpotter	21.79%	12.07%
MaskTextSpotter	69.43%	13.72%
CharNet	74.15%	35.53%
MRZNet	<b>100.00%</b>	<b>99.25%</b>

<sup>a</sup> Recognition rate are based on an average of 76 out of 88 characters available in the free version

Table 3: MRZ detection and character recognition (in macro-f1 score) results on MIDV-500 MRZ dataset [2] for MRZNet and other solutions

Method	MRZ Detection	Character Recog.
PassportEye	27.32%	64.93%
MRZ-Detection	46.30%	76.00%
UltimateMRZ <sup>a</sup>	<b>77.15%</b>	71.69%
TextSpotter	21.20%	13.37%
MaskTextSpotter	69.15%	19.42%
CharNet	74.18%	28.44%
MRZNet	73.94%	<b>85.18%</b>

<sup>a</sup> Recognition rate are based on an average of 76 out of 88 characters available in the free version

Table 4: MRZ detection and character recognition (in macro-f1 score) results on SyntheticMRZ [15] dataset for MRZNet and other solutions. For TextSpotter, MaskTextSpotter and CharNet, results can not be generated due to lack of ground truth bounding box label

Method	MRZ Detection	Character Recog.
PassportEye	46.87%	84.42%
MRZ-Detection	86.06%	87.59%
UltimateMRZ <sup>a</sup>	42.40%	78.40%
TextSpotter	NA	NA
MaskTextSpotter	NA	NA
CharNet	NA	NA
MRZNet	<b>88.66%</b>	<b>90.09%</b>

<sup>a</sup> Recognition rate are based on an average of 76 out of 88 characters available in the free version

### 5.3 Ablation Study

We performed ablation studies to evaluate the effectiveness of the two stage model and the ASPP layers, with the results are reported in Table 6 and Table 7. From Table 6, it can be inferred that using only MRZSpotter (coarse) will result in poor MRZ text recognition accu-

Table 5: Recognition speed comparison on our test set, (*mean*  $\pm$  *std*). GPU: a single GeForce RTX 3090; CPU: Intel(R) Xeon(R) Gold 5220R

Method	CPU time (s)	GPU time(s)
PassportEye	0.68 $\pm$ 0.45	
MRZ-Detection	2.46 $\pm$ 1.01	
UltimateMRZ	0.24 $\pm$ 0.16	0.14 $\pm$ 0.16
TextSpotter	23.12 $\pm$ 3.91	0.90 $\pm$ 0.49
MaskTextSpotter	6.74 $\pm$ 1.17	1.58 $\pm$ 0.77
CharNet	80.62 $\pm$ 9.53	9.25 $\pm$ 5.34
MRZNet	14.82 $\pm$ 0.97	0.51 $\pm$ 0.78

Table 6: Results on validation set from MRZSpotter (coarse). We show the variation of bounding box detection IOU and MRZ text recognition macro F1-score with different numbers of ASPP layers

Num. of ASPP	IoU	Macro F1-score
0	0.8701	56.56%
1	0.8508	67.87%

racy (with 67.87% macro f1-score as the best result). The primary reason is that the resolution of input image is low. However, we also found that the narrowing and box rotation angle range is an important factor since increasing the data augmentation rotation angle of MRZSpotter (fine) results in a worse accuracy. Including a single ASPP layer to MRZSpotter (coarse) improved the MRZ text recognition performance from 56.56% to 67.87%. However, the bounding box localization accuracy decreased from 0.8701 to 0.8508 IoU. Considering the objective of MRZSpotter (coarse) is localization, we haven't included any ASPP layers in the final model for MRZSpotter (coarse). From Table 7, it can be seen that the MRZ recognition accuracy is much improved by using MRZSpotter (fine) after MRZSpotter (coarse). Including one single layer of ASPP improved the MRZ text recognition accuracy from 98.40% to 98.91%. By stacking 3 ASPP layers, the accuracy further improved to 99.21%. These results demonstrate the impact of the proposed two stage model and the ASPP layers. In Table 8, we compared the different loss function for score map. From the results, it can be seen that the dice loss as adopted by our approach outperforms the balanced cross-entropy loss as adopted by EAST [57].

## 6 Conclusion

In this work, we presented MRZNet, a framework specifically designed for localizing and recognizing the MRZ

Table 7: Results on validation set from MRZSpotter (fine). We show the variation of bounding box detection IOU and MRZ text recognition macro F1-score with different numbers of ASPP layers

Num. of ASPP	IoU	Macro F1-score
0	0.9071	98.40%
1	0.9059	98.91%
3	0.9144	99.21%

Table 8: Results on validation set from MRZSpotter (fine). We compare results for balanced cross-entropy loss for score map as adopted by EAST [57] and dice loss as adopted by our study

loss	IoU	Macro F1-score
balanced cross-entropy	0.8906	98.34%
dice	0.9144	99.21%

text in passport and visa images. A novel two stage model process is adopted so that MRZNet can handle passport/visa images of varies sizes from high resolution images. We proposed MRZSpotter, an end-to-end network for detecting and recognizing MRZ text. By stacking multiple layers of ASPP, we increased the receptive field of the model and improved the MRZ text recognition accuracy. Experiment evaluation demonstrated the effectiveness of our approach compared with existing state-of-the-art models. Possible future research directions could include: 1) adding a dewarp component to the framework to make the pipeline robust to passport images that are warped with curved text lines; 2) modifying the architecture for single character level bounding box detection and recognition in order to further improve the overall robustness of the pipeline; 3) evaluating the performance of our models on passport/visa having MRZ region soiled by smoke, water/mud, ink or other artifacts.

## 7 Declaration

### 7.1 Funding

This research was supported by Lendbuzz.

### 7.2 Conflict of Interest

The authors declare that they have no conflict of interest.

### 7.3 Availability of data and material

Not available.

### 7.4 Code availability

Not available.

## References

1. Abhishek Dutta, A.G., Zisserman, A.: (2020). URL <https://www.robots.ox.ac.uk/~vgg/software/via/>
2. Arlazarov, V.V., Bulatov, K.B., Chernov, T.S., Arlazarov, V.L.: Midv-500: a dataset for identity document analysis and recognition on mobile devices in video stream **43**(5) (2019)
3. Bessmeltsev, V., Bulushev, E., Goloshevsky, N.: High-speed OCR algorithm for portable passport readers. 21st International Conference on Computer Graphics and Vision, GraphiCon'2011 - Conference Proceedings (2011)
4. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE Transactions on Pattern Analysis and Machine Intelligence **40**(4), 834–848 (2018). DOI 10.1109/tpami.2017.2699184. URL <http://dx.doi.org/10.1109/TPAMI.2017.2699184>
5. Chernyshova, Y.S., Aliev, M.A., Gushchanskaia, E.S., Sheshkus, A.V.: Optical Font Recognition in Smartphone-Captured Images, and its Applicability for ID Forgery Detection. Eleventh International Conference on Machine Vision (ICMV 2018) p. 59 (2019). DOI 10.1117/12.2522955. URL <http://arxiv.org/abs/1810.08016>. ArXiv: 1810.08016
6. Dai, Y., Huang, Z., Gao, Y., Xu, Y., Chen, K., Guo, J., Qiu, W.: Fused Text Segmentation Networks for Multi-oriented Scene Text Detection. arXiv:1709.03272 [cs] (2018). URL <http://arxiv.org/abs/1709.03272>. ArXiv: 1709.03272
7. Deng, D., Liu, H., Li, X., Cai, D.: PixelLink: Detecting Scene Text via Instance Segmentation. arXiv:1801.01315 [cs] (2018). URL <http://arxiv.org/abs/1801.01315>. ArXiv: 1801.01315
8. Deng, J., Dong, W., Socher, R., Li, L., Kai Li, Li Fei-Fei: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255 (2009)
9. Deng, L., Gong, Y., Lin, Y., Shuai, J., Tu, X., Zhang, Y., Ma, Z., Xie, M.: Detecting Multi-Oriented Text with Corner-based Region Proposals. Neurocomputing **334**, 134–142 (2019). DOI 10.1016/j.neucom.2019.01.013. URL <http://arxiv.org/abs/1804.02690>. ArXiv: 1804.02690
10. Donoser, M., Arth, C., Bischof, H.: Detecting, Tracking and Recognizing License Plates. In: Y. Yagi, S.B. Kang, I.S. Kweon, H. Zha (eds.) Computer Vision – ACCV 2007, Lecture Notes in Computer Science, pp. 447–456. Springer, Berlin, Heidelberg (2007). DOI 10.1007/978-3-540-76390-1\_44
11. doubango.org: (2020). URL <https://github.com/DoubangoTelecom/ultimateMRZ-SDK#Getting-started-Adding-the-SDK-to-your-project>

12. Fabrizio, J., Marcotegui, B., Cord, M.: Text segmentation in natural scenes using toggle-mapping. In: 2009 16th IEEE International Conference on Image Processing (ICIP), pp. 2373–2376 (2009)
13. Giusti, A., Ciresan, D.C., Masci, J., Gambardella, L.M., Schmidhuber, J.: Fast image scanning with deep max-pooling convolutional neural networks. 2013 IEEE International Conference on Image Processing (2013). DOI 10.1109/icip.2013.6738831. URL <http://dx.doi.org/10.1109/ICIP.2013.6738831>
14. González, Á., Bergasa, L.M., Yebes, J.J.: Location in complex images (2012)
15. Hartl, A., Arth, C., Schmalstieg, D.: Real-time Detection and Recognition of Machine-Readable Zones with Mobile Devices. In: Proceedings of the 10th International Conference on Computer Vision Theory and Applications, pp. 79–87. SCITEPRESS - Science and Technology Publications, Berlin, Germany (2015)
16. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask R-CNN. arXiv:1703.06870 [cs] (2018). URL <http://arxiv.org/abs/1703.06870>. ArXiv: 1703.06870
17. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016). DOI 10.1109/cvpr.2016.90. URL <http://dx.doi.org/10.1109/cvpr.2016.90>
18. He, P., Huang, W., He, T., Zhu, Q., Qiao, Y., Li, X.: Single Shot Text Detector with Regional Attention. arXiv:1709.00138 [cs] (2017). URL <http://arxiv.org/abs/1709.00138>. ArXiv: 1709.00138
19. He, T., Tian, Z., Huang, W., Shen, C., Qiao, Y., Sun, C.: An end-to-end textspotter with explicit alignment and attention. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (2018). DOI 10.1109/cvpr.2018.00527. URL <http://dx.doi.org/10.1109/CVPR.2018.00527>
20. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation 9(8), 1735–1780 (1997)
21. Hu, H., Zhang, C., Luo, Y., Wang, Y., Han, J., Ding, E.: WordSup: Exploiting Word Annotations for Character based Text Detection. arXiv:1708.06720 [cs] (2017). URL <http://arxiv.org/abs/1708.06720>. ArXiv: 1708.06720
22. Karatzas, D., Gomez-Bigorda, L., Nicolaou, A., Ghosh, S., Bagdanov, A., Iwamura, M., Matas, J., Neumann, L., Chandrasekhar, V.R., Lu, S., Shafait, F., Uchida, S., Valveny, E.: ICDAR 2015 competition on Robust Reading. In: 13th IAPR International Conference on Document Analysis and Recognition, ICDAR 2015 - Conference Proceedings, pp. 1156–1160. IEEE Computer Society (2015). DOI 10.1109/ICDAR.2015.7333942
23. Kasar, T., Ramakrishnan, A.: Multi-script and Multi-oriented Text Localization from Scene Images. pp. 1–14 (2012). DOI 10.1007/978-3-642-29364-1\_1
24. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization (2014)
25. Kostro, D., Zasso, M.: (2020). URL <https://github.com/image-js/mrz-detection>
26. Lee, H., Kwak, N.: Character recognition for the machine reader zone of electronic identity cards. In: 2015 IEEE International Conference on Image Processing (ICIP), pp. 387–391 (2015). DOI 10.1109/ICIP.2015.7350826
27. Liao, M., Shi, B., Bai, X.: TextBoxes++: A Single-Shot Oriented Scene Text Detector. IEEE Transactions on Image Processing 27(8), 3676–3690 (2018). DOI 10.1109/TIP.2018.2825107. URL <http://arxiv.org/abs/1801.02765>. ArXiv: 1801.02765
28. Liao, M., Wan, Z., Yao, C., Chen, K., Bai, X.: Real-time Scene Text Detection with Differentiable Binarization. arXiv:1911.08947 [cs] (2019). URL <http://arxiv.org/abs/1911.08947>. ArXiv: 1911.08947
29. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C.: Ssd: Single shot multibox detector. Lecture Notes in Computer Science p. 21–37 (2016). DOI 10.1007/978-3-319-46448-0\_2. URL [http://dx.doi.org/10.1007/978-3-319-46448-0\\_2](http://dx.doi.org/10.1007/978-3-319-46448-0_2)
30. Liu, X., Liang, D., Yan, S., Chen, D., Qiao, Y., Yan, J.: FOTS: Fast Oriented Text Spotting with a Unified Network. arXiv:1801.01671 [cs] (2018). URL <http://arxiv.org/abs/1801.01671>. ArXiv: 1801.01671
31. Liu, X., Liang, D., Yan, S., Chen, D., Qiao, Y., Yan, J.: Fots: Fast oriented text spotting with a unified network. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 5676–5685 (2018)
32. Liu, Y., He, T., Chen, H., Wang, X., Luo, C., Zhang, S., Shen, C., Jin, L.: Exploring the Capacity of an Orderless Box Discretization Network for Multi-orientation Scene Text Detection. arXiv:1912.09629 [cs] (2020). URL <http://arxiv.org/abs/1912.09629>. ArXiv: 1912.09629
33. Liu, Z., Sarkar, S.: Robust Outdoor Text Detection Using Text Intensity and Shape Features
34. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3431–3440 (2015)
35. Long, S., Ruan, J., Zhang, W., He, X., Wu, W., Yao, C.: TextSnake: A Flexible Representation for Detecting Text of Arbitrary Shapes. arXiv:1807.01544 [cs] (2018). URL <http://arxiv.org/abs/1807.01544>. ArXiv: 1807.01544
36. Lyu, P., Liao, M., Yao, C., Wu, W., Bai, X.: Mask TextSpotter: An End-to-End Trainable Neural Network for Spotting Text with Arbitrary Shapes. arXiv:1807.02242 [cs] (2018). URL <http://arxiv.org/abs/1807.02242>. ArXiv: 1807.02242
37. Lyu, P., Liao, M., Yao, C., Wu, W., Bai, X.: Mask textspotter: An end-to-end trainable neural network for spotting text with arbitrary shapes. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 67–83 (2018)
38. Lyu, P., Yao, C., Wu, W., Yan, S., Bai, X.: Multi-Oriented Scene Text Detection via Corner Localization and Region Segmentation. arXiv:1802.08948 [cs] (2018). URL <http://arxiv.org/abs/1802.08948>. ArXiv: 1802.08948
39. Merino-Gracia, C., Lenc, K., Mirmehdi, M.: A Head-Mounted Device for Recognizing Text in Natural Scenes. In: M. Iwamura, F. Shafait (eds.) Camera-Based Document Analysis and Recognition, Lecture Notes in Computer Science, pp. 29–41. Springer, Berlin, Heidelberg (2012). DOI 10.1007/978-3-642-29364-1\_3
40. Minetto, R., Thome, N., Cord, M., Stolfi, J., Príncipe, F., Guyonard, J., Leite, N.: Text detection and recognition in urban scenes. In: 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops), pp. 227–234 (2011). DOI 10.1109/ICCVW.2011.6130247
41. Mishra, A., Alahari, K., Jawahar, C.: Scene Text Recognition using Higher Order Language Priors. In: Proceedings of the British Machine Vision Conference 2012, pp. 127.1–127.11. British Machine Vision Association, Surrey (2012). DOI 10.5244/C.26.127. URL <http://www.bmva.org/bmvc/2012/BMVC/paper127/index.html>
42. Neumann, L., Matas, J.: Real-time scene text localization and recognition. pp. 3538–3545 (2012). DOI 10.1109/CVPR.2012.6248097

43. 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* **20**(3), 800–813 (2011). DOI 10.1109/TIP.2010.2070803. Conference Name: IEEE Transactions on Image Processing
44. Petrova, O., Bulatov, K.: Methods of machine-readable zone recognition results post-processing. In: Eleventh International Conference on Machine Vision (ICMV 2018), vol. 11041, p. 110411H. International Society for Optics and Photonics (2019). DOI 10.1117/12.2522792
45. SakuraRiven: (2020). URL <https://github.com/SakuraRiven/EAST>
46. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C.: Mobilenetv2: Inverted residuals and linear bottlenecks. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (2018). DOI 10.1109/cvpr.2018.00474. URL <http://dx.doi.org/10.1109/CVPR.2018.00474>
47. Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., LeCun, Y.: Overfeat: Integrated recognition, localization and detection using convolutional networks (2013)
48. Shi, B., Bai, X., Belongie, S.: Detecting Oriented Text in Natural Images by Linking Segments. arXiv:1703.06520 [cs] (2017). URL <http://arxiv.org/abs/1703.06520>. ArXiv: 1703.06520
49. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition (2014)
50. Smith, R.: An overview of the tesseract ocr engine. In: Ninth international conference on document analysis and recognition (ICDAR 2007), vol. 2, pp. 629–633. IEEE (2007)
51. Tretjakov, K.: PassportEye: Extraction of machine-readable zone information from passports, visas and id-cards via OCR (2016). URL <https://github.com/konstantint/PassportEye>
52. Wang, J., Hu, X.: Gated recurrent convolution neural network for ocr. In: Advances in Neural Information Processing Systems, pp. 335–344 (2017)
53. Wang, W., Xie, E., Li, X., Hou, W., Lu, T., Yu, G., Shao, S.: Shape Robust Text Detection with Progressive Scale Expansion Network. arXiv:1903.12473 [cs] (2019). URL <http://arxiv.org/abs/1903.12473>. ArXiv: 1903.12473
54. Xing, L., Tian, Z., Huang, W., Scott, M.R.: Convolutional Character Networks. arXiv:1910.07954 [cs] (2019). URL <http://arxiv.org/abs/1910.07954>. ArXiv: 1910.07954
55. Xu, Y., Duan, J., Kuang, Z., Yue, X., Sun, H., Guan, Y., Zhang, W.: Geometry Normalization Networks for Accurate Scene Text Detection. arXiv:1909.00794 [cs] (2019). URL <http://arxiv.org/abs/1909.00794>. ArXiv: 1909.00794
56. Zhang, Z., Zhang, C., Shen, W., Yao, C., Liu, W., Bai, X.: Multi-Oriented Text Detection with Fully Convolutional Networks. arXiv:1604.04018 [cs] (2016). URL <http://arxiv.org/abs/1604.04018>. ArXiv: 1604.04018
57. Zhou, X., Yao, C., Wen, H., Wang, Y., Zhou, S., He, W., Liang, J.: EAST: An Efficient and Accurate Scene Text Detector. arXiv:1704.03155 [cs] (2017). URL <http://arxiv.org/abs/1704.03155>. ArXiv: 1704.03155 version: 2
58. Zhu, K.h., Qi, F.h., Jiang, R.j., Xu, L.: Automatic character detection and segmentation in natural scene images. *Journal of Zhejiang University: Science A* **8**, 63–71 (2007). DOI 10.1631/jzus.2007.A0063

## Author Biographies

**Yichuan** received Ph.D. degree in Biomedical Engineering from Drexel University in 2018. He is currently working as a machine learning engineer at Lendbuzz in Boston, Massachusetts, where his research includes document information extraction and document quality estimation and correction. Before joining Lendbuzz, he worked at Harvard Medical School where he developed deep learning algorithms to classify and segment medical images.



**Hailey James** received Bachelor's degree in Computer Science from Harvard College in Cambridge, Massachusetts in 2018. She is currently working as a machine learning engineer at Lendbuzz in Boston, Massachusetts, where her research includes image and document forensics. Her interests also include work in fairness, human-computer cooperation, and explainability in artificial intelligence.



**Otkrist Gupta** is currently Vice President of Data Science at Lendbuzz, focussing on deep learning with applications in finance and computer vision. He completed his Ph.D. at MIT Media Lab from camera culture group. His research is focused on inventing new algorithms for deep learning for health screening and diagnosis, hidden geometry detection, exploiting techniques from optimization, linear algebra and compressive sensing. He also works on designing algorithms for futuristic 3D projective displays. Before joining MIT Media Lab Otkrist worked in Google Now team where he built voice actions such as take a picture and what's on my Chromecast and worked on voice response quality from Google Now. He also worked at LinkedIn where he developed services such as Smart ToDo, Ultra fast auto-complete, Notifications and CheckIn platform. He completed his bachelors from Indian Institute of Technology Delhi (IITD) in Computer Science with emphasis on algorithms and linear algebra. After graduating from IITD, he worked for one year in field of High Frequency Trading at Tower Research Capital.



**Dan Raviv** is the Chief Technology Officer of Lenbuzz Inc, an AI-based fintech underwriting company, and is faculty in the department of Engineering, Tel Aviv University, Israel. Dan did his post-doc at MIT in the Media Lab, working on various problems in the intersection of Geometry, Computer vision, and Machine learning, and is one of the leading researchers in Geometric Deep Learning. Dan was awarded the 2016 biennial award for Imaging Sciences granted by SIAM and published dozens of academic papers in his research field. Dan earned his Ph.D. and M.Sc. in computer science from the Technion – Israel Institute of Technology, Israel, and holds a bachelor’s degree in Mathematics and Computer Science, Summa Cum Lauda, from the Technion as well. Dan was a member of the Technion’s Excellence program, which handpicks the best and the brightest.

