Scene Text Image Super-resolution in the wild

Wenjia Wang*[1], Enze Xie*[2], Xuebo Liu[1],
Wenhai Wang[3], Ding Liang[1], Chunhua Shen**[4], and Xiang Bai[5]

[1] SenseTime Research [2] The University of Hong Kong
[3] Nanjing University [4] The University of Adelaide of Science and Technology [5] Huazhong University
wangwenjia@sensetime.com

1 Extensive Experiments on Our Method

1.1 Synthetic LR vs. TextZoom LR

To demonstrate the superiority of paired scene text SR images, we compare the performance of the models trained on synthetic datasets and our TextZoom dataset. Traditional SISR tasks simply down-sample HR image by bicubic interpolation to generate corresponding LR images. To illustrate the superiority of real LR over synthetic LR, we train our model on the bicubic down-sampled LR images and real LR images to show the performance.

Table 1. The comparison of the models trained on synthetic LR and real LR. The listed results are the models evaluated on proposed TextZoom LR images. For better displaying, we calculated the average accuracy. The recognition accuracies are tested by the official released model of ASTER [14], MORAN [11] and CRNN [13]. 'Syn' denotes down-sampled LR and 'Real' denotes proposed LR images.

Method	train data	Accuracy of ASTER [14]				Accuracy of MORAN [11]				Accuracy of CRNN [13]			
		easy	medium	hard	**average**	easy	medium	hard	**average**	easy	medium	hard	**average**
BICUBIC	–	64.7%	42.4%	31.2%	47.2%	60.6%	37.9%	30.8%	44.1%	36.4%	21.1%	21.1%	26.8%
SRResNet	Syn	66.4%	44.4%	32.4%	48.9%	**61.8%**	39.6%	31.0%	45.2%	37.4%	21.6%	21.2%	27.3%
	Real	**69.4%**	**47.3%**	**34.3%**	**51.3%**	60.7%	**42.9%**	**32.6%**	**46.3%**	**39.7%**	**27.6%**	**22.7%**	**30.6%**
LapSRN	Syn	66.5%	43.9%	32.2%	48.7%	61.8%	39.0%	30.7%	44.9%	37.5%	21.8%	20.9%	27.3%
	Real	**71.5%**	**48.6%**	**35.2%**	**53.0%**	**64.6%**	**44.9%**	**32.2%**	**48.3%**	**46.1%**	**27.9%**	**23.6%**	**33.3%**
TSRN(ours)	Syn	67.5%	45.3%	33.0%	49.7%	61.7%	40.4%	30.6%	45.3%	37.8%	22.0%	21.0%	27.6%
	Real	**75.1%**	**56.3%**	**40.1%**	**58.3%**	**70.1%**	**53.3%**	**37.9%**	**54.8%**	**52.5%**	**38.2%**	**31.4%**	**41.4%**

We selected SRResNet [9], LapSRN [7] and our proposed method TSRN, and trained them on the synthetic LR and real LR datasets for a 2X model respectively. We trained 6 models in all and evaluated them on our proposed TextZoom subsets. From Table 1, we can figure that the three methods trained on real LR (TextZoom) dataset outperform the models trained on synthetic LR

* Equal Contribution.
** Corresponding Author.

obviously in accuracy. For our TSRN, the model trained on real LR could surpass the synthetic LR for nearly 9.0% on ASTER and MORAN, and nearly 14.0% on CRNN.

1.2 Speed & Accuracy.

In this task, we take the recognition accuracy as the most import evluation metric. To figure out whether it is wise to increase the accuracy at the cost of the extra computation consumption of TSRN, we compare the number of parameter, FLOPs and inference FPS of w and w/o super-resolution. The inference FPS means the FPS of recognizing the text images w or w/o SR. Through Table 2, we can find that the proposed method is relatively tiny compared to the recognition network. The FPS of 'with TSRN' is nearly equal to direct recognition of attention based recognizer ASTER [14] and MORAN [11]. The FPS of CTC based recognizer CRNN decreases when adding the TSRN, but the improvement of accuracy is very considerable. So it would be a suitable manipulation to take super-resolution as a pre-processing procedure before recognition. (All of the FPSs were tested on a single GTX 1080Ti GPU with the same batch-size of 50.)

Table 2. Computation and speed comparison between w or w/o super resolution when recognize TextZoom. '×' means directly recognizing BICUBIC up-sampled LR images. '√' means recognizing after super-resolving images by our TSRN. **The inference FPS means the FPS of recognizing w or w/o SR.**

Computation Cost Analysis					
Recognizer	TSRN(ours)	Average Accuracy	FLOPs	Parameters	**Inference FPS**
ASTER [14]	×	47.2%	4.72G	20.99M	21.97
	√	58.3% (+10.1%)	4.72G + 0.72G	20.99M + 2.8M	21.67
MORAN [11]	×	44.1%	0.73G	20.3M	63.2
	√	54.8% (+10.7%)	0.73G + 0.72G	20.3M+2.8M	59.6
CRNN [13]	×	26.8%	0.64G	8.3M	514.7
	√	41.4% (+14.6%)	0.64G + 0.72G	8.3M + 2.8M	340.6

1.3 Binary Mask

In text images, the characters are usually in a unified color. The only texture information is the character color and background color. For brevity, we concatenate the binary mask with text images as input (Fig. 1). The character regions render 1 and the background regions render 0. This input can be viewed as a transcendental semantic segmentation label of text images since most of the text images only contain 2 colors: the text color and background color. The masks are simply generated by calculating the average gray scale of the RGB images.

Fig. 1. The demonstration of the binary masks.

Table 3. The ablation study of binary masks.

Ablation Study of Masks				
Configuration	Mask	Accuracy		
		easy	medium	hard
5 × SRB	×	73.9%	51.6%	36.0%
	√	**74.5%**	**53.3%**	**37.3%**

1.4 Discussion about SRB

To build the best architecture of SRB, we gradually modify this two essential configuration: the number of hidden units and the number of blocks. Our method select 5 × SRB with 32 hidden units each. In this section, we do ablation study on this two component separately.

1) Hidden Units. The BLSTMs are used to build sequence dependence in the text lines, so we hypothesize that more hidden units could get better performance. By the experiments, we compare 0, 16, 32, 64, 128 of hidden layers. 0 Hidden Units represents SRResNet. The results demonstrate that the network would achieve best accuracy when the number of hidden unit equal 32 (Table 4) Too many hidden units achieve lower performance since it already build the sequence-dependence well.

2) Block Number. To figure out whether we can achieve better performance by building deeper network, we stack different number of SRBs to compare the performance. In Table 5, we compare our method with 4, 5, 6, 7 SRBs. We can find that more SRBs may not boost up the performance. The accuracy of 7 SRBs even decrease obviously. Stacking 5 SRBs, the network saturates and could get the best performance.

Our configuration of Sequence Residual Block is then shown in Table 6.

Table 4. Comparison between different number of hidden units of our proposed method on TextZoom.

Ablation Study of Hidden Units				
Configuration		Accuracy		
SRBs	Hidden Units	easy	medium	hard
5	0	69.6%	48.3%	34.3%
	16	71.6%	52.1%	36.3%
	32	**74.5%**	**53.3%**	**37.3%**
	64	71.9%	50.8%	35.8%
	128	71.4%	47.3%	33.1%

Table 5. Comparison between different number of SRBs of our proposed method on TextZoom.

Ablation Study of SRBs				
Configuration		Metrics		
SRBs	Hidden Units	easy	medium	hard
4	32	73.3%	52.1%	35.8%
5		**74.5%**	**53.3%**	**37.3%**
6		74.1%	52.7%	37.0%
7		72.3%	50.9%	35.6%

1.5 PSNR & SSIM

To calculate the PSNR[dB] and SSIM, we borrow the code from https://github.com/open-mmlab/mmsr. From Table 7, our PSNR of medium and hard subsets are not so good because PSNR is pixel-to-pixel calculated, while SSIM is calculated with a 11×11 sliding kernel. The central alignment module would introduce slight pixel shift so the PSNR is somewhat lower than other SR methods. Usually, PSNR and SSIM could not represent the visual quality of the images [9], in this task, it is also not so important compared to accuracy.

2 Central Alignment Module

Our central alignment module is based on Spatial Transformation Network [4]. The network predicts a set of control points and then then image is rectified

Table 6. Network configuration summary. The first row is the top layer. 'k', 's' and 'p' stand for kernel size, stride and padding size respectively.

Type	Configurations
FeatureMap	B×64×Height×Wdith
Convolution	#maps:64, k:3×3, s:1 p:1
BatchNormalization	
PReLU	
Convolution	#maps:64, k:3×3, s:1 p:1
BatchNormalization	
Convolution	#maps:64, k:1×1, s:1 p:0
Permutuation	
Bi-LSTM	#hidden_units: 32
Map-to-Sequence	
Permutuation	
Bi-LSTM	#hidden_units: 32
Map-to-Sequence	
Permutuation	
Short Cut Connection	
FeatureMap	B×64×Height×Wdith

by a Thin-Plate-Spline(TPS) [1] transformation. Our central alignment module mainly use horizontal or vertical shift. But sometimes the background region need different transformation scale to let the character region more central placed. So we use TPS transformation here to let the transformation flexible. As shown in Figure. 2, the transformation is different between different points.

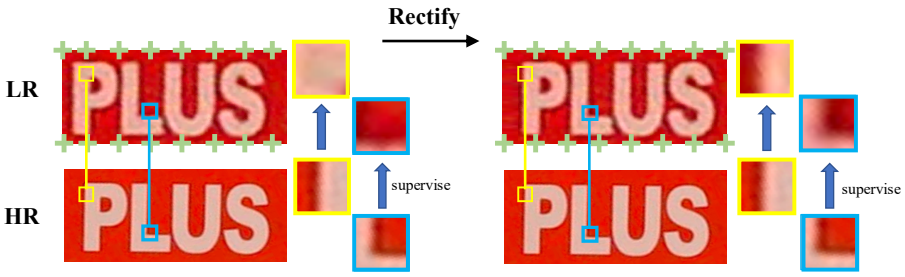


Fig. 2. Demonstration of central alignment module.

Table 7. PSNR and SSIM results of different SR methods on TextZoom.

Method	Loss Function	PSNR			SSIM		
		easy	medium	hard	easy	medium	hard
BICUBIC	–	22.35	18.98	19.39	0.7884	0.6254	0.6592
SRCNN [3]	L_2	23.48	**19.06**	19.34	0.8379	0.6323	0.6791
VDSR [6]	L_2	24.62	18.96	**19.79**	0.8631	0.6166	0.6989
SRResNet [9]	$L_2 + L_{tv} + L_p$	24.36	18.88	19.29	0.8681	0.6406	0.6911
RRDB [8]	L_1	22.12	18.35	19.15	0.8351	0.6194	0.6856
EDSR [10]	L_1	24.26	18.63	19.14	0.8633	0.6440	0.7108
RDN [16]	L_1	22.27	18.95	19.70	0.8249	0.6427	0.7113
LapSRN [7]	*Charbonnier*	24.58	18.85	19.77	0.8556	0.6480	0.7087
TSRN(ours)	$L_2 + L_{GP}$	**25.07**	18.86	19.71	**0.8897**	**0.6676**	**0.7302**

2.1 Performance on Manual Enlarged Misalignment

We can find from ablation study that the central alignment module could improved the average accuracy for less than 2.0%. Indeed, it can perform better on more misaligned text image pairs. To prove that, we do data augmentation aiming at generating more misaligned image pairs. We crop our dataset TextZoom using a box with a 90% width and 90% height of the original image size randomly slide on the LR image, and get a region of 90%×90% image. The HR images are not cropped. We train on the cropped dataset and evaluate on TextZoom. In Table 8, we show the performance of central alignment module on our manual cropped misalignment data. From the results in Table.8, we can find that the accuracy could be sharply improved.

Table 8. Performance of w or w/o central alignment module on TextZoom which was trained on the mannual enlarged misaligned data.

Method	Accuracy			
	easy	medium	hard	average
5×SRB	66.8%	50.0%	35.0%	51.6%
5×SRB+Align	74.4%	55.6%	38.8%	57.4%
Improvement	+7.6%	+5.6%	+3.8%	+5.8%

We show visulization results in Fig. 3. The third row is the SR images trained without alignment. We can find that the double shadow and artifacts are very serve when trained without central alignment module. We can find that many words are still correctly recognized even with strong double shadow.

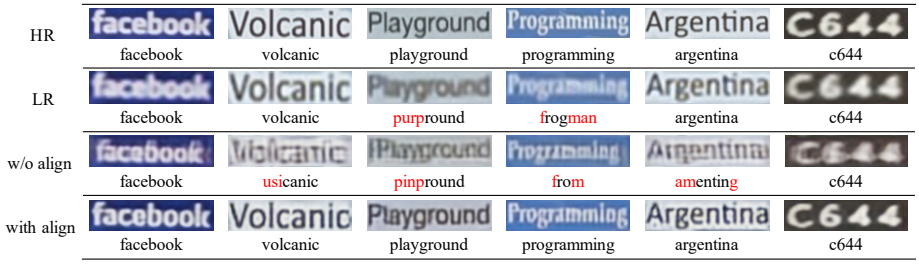


Fig. 3. Comparison of w or w/o central align on enlarged misaligned data. The character strings under the images are the recognition results tested by ASTER [14]. Those in red means wrongly recognition. For better display, we crop some obvious patches to compare the performance of w or w/o alignment.

2.2 Plugged into Other SR methods

In this study, we compare the performance of w or w/o central alignment module on our dataset. (Table 9). We display the performance of six models: w and w/o Central Alignment Module on SRResNet, LapSRN and ours seperately. The improvement of central align on these methods illustrate that it is a conveniently pluggable module for SR networks, and all the performance could be improved.

Table 9. Comparison between w or w/o Central Alignment Module on TextZoom.

Method	Alignment	Accuracy			
		easy	medium	hard	average
SRResNet	×	69.6%	47.6%	34.3%	51.7%
	√	**70.0%**	**49.6%**	**36.0%**	**53.0%**
LapSRN	×	71.5%	48.6%	35.2%	53.0%
	√	**71.7%**	**50.3%**	**35.7%**	**53.7%**
5 × SRB	×	74.5%	53.3%	37.3%	56.2%
	√	**74.8%**	**55.7%**	**39.6%**	**57.8%**

2.3 Comparison with CoBi Loss.

CoBi Loss was proposed in [15] to tackle the misalignment. It is based on Contextual Loss [12]. It modified the nearest neighbor search and considers local contextual similarities with weighted spatial awareness. The CoBi Loss used pre-trained VGG-19 features and select several conv layers as deep features. Its formulation is shown in Eqn. 1 2 3. The results are shown in Table 10. It is less

practical in this task because the pre-trained model is trained on a classification dataset.

$$CoBi(P,Q) = \frac{1}{N}\sum_{i}^{N} \underset{j=1,...M}{min}(\mathbb{D}_{p_i,q_j} + \omega_s \mathbb{D}^{'}_{p_i,q_j}) \tag{1}$$

$$\mathbb{D}^{'}_{p_i,q_j} = ||(x_i, y_i) - (x_i, y_i)||_2 \tag{2}$$

$$CoBiLoss = CoBi_{RGB}(P,Q,n) + \lambda CoBi_{VGG}(P,Q) \tag{3}$$

Table 10. Comparison between CoBi Loss and central alignment module.

Method	Accuracy		
	easy	medium	hard
CoBi Loss	74.0%	51.6%	36.0%
L_2 + alignment	74.8%	55.7%	39.6%

3 Detailed Information of TextZoom.

3.1 Annotation of SR-RAW and RealSR.

SR-RAW [15] is collected by seven different focal lengths with SONY FE camera, ranging from 24-240mm. We demonstrate it in Fig. 4. There are totally 500 images in SR-RAW dataset, where 450 in train set and 50 in test set. The images are then aligned via field of view (FOV) matching and geometric transformation. The images captured in shorted focal lengths could be used as LR images while those captured in longer lengths as corresponding ground-truth. The author of SR-RAW [15] applied down-sample operation as offset when the ratio does not match precisely. For example, when use (35mm, 150mm) pairs to train a 4X model, the 150mm images should be down-sampled to 140mm at first. In our project, we follow this strategy in our dataset pre-processing. We annotate all the images taken from 240mm focal length which contains recognizable text in SR-RAW dataset. AS showed in Fig. 4, the focal length decreases from left to right, from 240mm to 24mm. The smaller the focal length, the smaller the field of view. The annotated text images have the same text contexts but different resolutions. We display three groups in Fig. 4: 'STAR', 'QUEST', '510-401-4657'. In this image, the text images cropped from 35mm and 24mm are hardly recognizable. How many clear images in a group of 7 images mainly depends on the height of original box in the 240mm focal lengths images.

In Table 11, we show the information of the cropped text images in SR-RAW. In the original images, Some groups of images do not have the 7th image, so the number of 24mm is less than the others. Through the table we can figure out

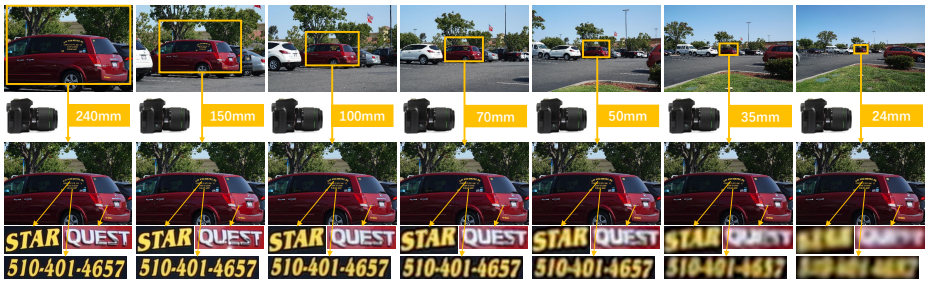


Fig. 4. The demonstration of the SR-RAW paired images and how we cropped text images.

that the recognition accuracy decreases obviously as the resolution degrades. We use the released ASTER [14] model to test the accuracy.

Table 11. The detailed information of the text images cropped from SR-RAW dataset. The 2nd to 7th groups of text images are cropped following the annotated bounding box in the 1st group.

Text Images in Train Set of SR-RAW							
Focal Length	240mm	150mm	100mm	70mm	50mm	35mm	24mm
Original Image Number	393	393	393	393	393	393	365
Text Box Number	9160	9160	9160	9160	9160	9160	8119
Recognition Accuracy	81.4%	69.0%	52.1%	38.6%	25.7%	15.0%	7.9%
Text Images in Test Set of SR-RAW							
Original Image Number	50	50	50	50	50	50	48
Text Box Number	1734	1734	1734	1734	1734	1734	1630
Recognition Accuracy	72.4%	65.3%	54.4%	35.6%	23.2%	13.6%	6.3%

RealSR [2] is captured by two Digital Single Lens Reflex(DSLR) cameras: Canon 5D3 and Nikon D810 with four focal lengths: 105mm, 50mm, 35mm, and 28mm. In RealSR [2], the images taken by 105mm focal length are used to generate HR images, while images taken by 50mm, 35mm, 28mm are used to generate 2X, 3X, 4X LR images separately. separately. For convenience, we only crop the 105mm, 50mm and 28mm. The non-horizontal text images are rotated to the most suitable angle for recognition (see Fig. 5).

In Table 12, we briefly show the statistics of text images in RealSR.

Align. In RealSR, the author aligned the image pairs by introducing a pixel-wise registration algorithm which take luminance difference into consideration. In SR-RAW [15], the Euclidean motion model is used as the pre-processing procedure. During training, a contextual bilateral loss is proposed to leverage the misalignment, but a pre-trained model is needed, and it brings high computa-

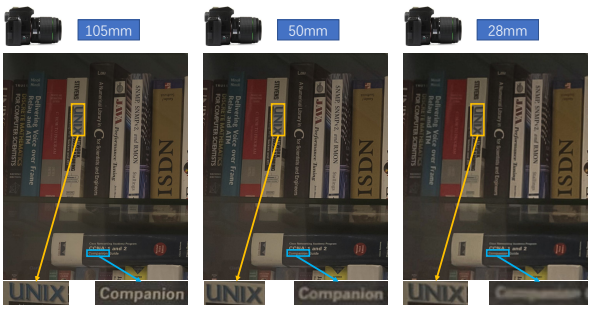


Fig. 5. The demonstration of the strategy we annotated the RealSR.

Table 12. The detailed information of the text images cropped from RealSR dataset.

Text Images in RealSR			
Focal Length	105mm	50mm	28mm
Original Number	115	115	115
Text Box Number	6048	6048	6048
Recognition Accuracy	75.0%	46.1%	16.7%

tion consumption. We adapted their proposed pre-processing method to align the original images and cropped our dataset following our annotation principal. While in training, we used central alignment module as replacement.

Accuracy by Height. The size of the cropped text boxes is diversed, We can figure that with the similar focal length, the accuracy of text images in RealSR is much higher than that in SR-RAW (Table 12 11). This mainly due to that the SR-RAW images are taken from a longer distance. So it is suitable to allocate images cropped from RealSR as subset **easy**.

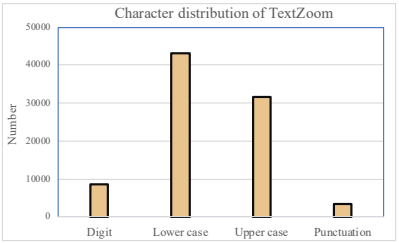
We divided the previous cropped images by height and found that the accuracy is relatively good when the height reaches 16-32 pixels, which is showed in Table 13. The images sized in (16-32) and (8-16) claim the majority in all the groups.

The accuracy of the images smaller than 8 pixels are too low, which hardly have any value for restoration. The images are hardly recognizable, so we discard the images the height of which is less than 8 pixels. (8-16, 16-32) should be a good pair to form a 2X train set for STR super-resolution task. For example, the text images taken from 150mm focal length and height sized in 16-32 pixels would be taken as a ground-truth for the 70mm counterpart. So we selected all the images the height of which range from 16 pixels to 32 pixels as our ground-truth image and up-sample them to the size of 128×32 (width×height), and the corresponding 2X LR images to the size of 64×16 (width×height).

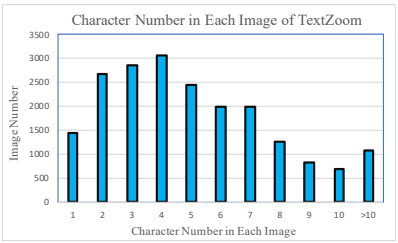
Table 13. The recognition accuracy of the text images divided by height.

Recognition Accuracy of images in different height							
Height(pixels)	128−	64−128	32−64	16−32	8−16	4−8	0−4
Number	1586	3957	9663	14862	15434	11866	5711
Recognition Accuracy	75.2%	84.2%	84.6%	79.5%	39.1%	2.8%	0.3%

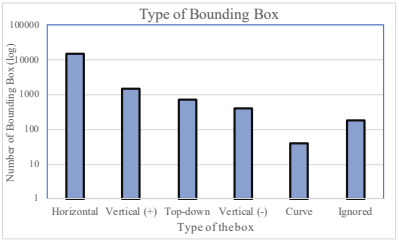
3.2 Statistical information.



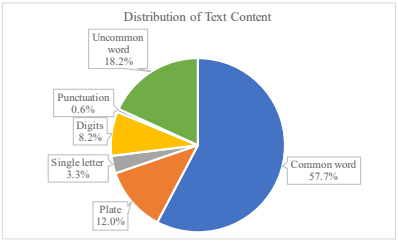
(a) Character distribution.



(b) Character number in each image.



(c) Direction of the bounding boxes. For better display, we list the logarithm of the ordinate number.



(d) Distribution of text contents in TextZoom.

Fig. 6. Statistical information of TextZoom.

We display some useful statistical information in Fig. 6. (a) Our dataset contains abundant characters and digits, including some punctuation. (b) Most of the lengths of the words range from 1-8 characters. (c) There are many randomly-placed boxes and books in the original images, so we count the direction type of the bounding boxes we annotated. 'Horizontal' means that the text image is horizontal placed, easy to read. 'Vertical(+)' denotes the text image is vertical and it should be rotated following the clockwise direction for 90 degrees, while 'Vertical(-)' denotes following anti-clockwise direction for 90 degrees. 'Top-down' denotes that the text image should be rotated 180 degrees for the best recognition. 'Curve' denote the text image is curved. 'Ignored' means that the

text is illegal (not digits, English letters or punctuation). (d) Via the generic lexicon which has 90k common words used in ICDAR2015 [5], we figure that 57.5% of the text contents are common English words. Plate includes car license plates, door number plates or street signs. They are the combination of digits, punctuation and letters. This kind of text account for 12% because there are many street views in the original images. Uncommon word claims 18.2% in all the texts. This kind of text are mainly rare words, phrases or compound words. Other meaningless strings like punctuation, single letter and digits account for the rest.

3.3 Task Analysis

Our dataset is challenging mainly for two reasons: the misalignment and ambiguity. Misalignment is unavoidable during data capture when the lens zoom in and out. Any slight camera movement could cause tens of pixels shift, especially the short focal lengths. And the pre-processing procedure cannot totally eliminate misalignment. We display some example images in Fig. 7.

From Fig. 7, we can figure that the misalignment varies and no specific regulation can be found since we do not have pixel-level annotation of the word location. The three different subsets are allocated appropriately by the difficulty. The misalignment and ambiguity becomes server as the difficulty increases. Note that the characters in HR images tend to locate in the center compared to those in LR. This mainly owing to that when we annotated the HR images, we artificially keep the text boxes at the centre of the images.

(a) Example images of easy subset.

(b) Example images of medium subset.

(c) Example images of hard subset.

Fig. 7. Demonstration of the images in TextZoom. The misalignment and ambiguity becomes server as the difficulty increases.

References

1. Bookstein, F.L.: Principal warps: Thin-plate splines and the decomposition of deformations. IEEE Trans. Pattern Anal. Mach. Intell. (1989)
2. Cai, J., Zeng, H., Yong, H., Cao, Z., Zhang, L.: Toward real-world single image super-resolution: A new benchmark and a new model. In: ICCV (2019)
3. Dong, C., Loy, C.C., He, K., Tang, X.: Image super-resolution using deep convolutional networks. TPAMI (2015)
4. Jaderberg, M., Simonyan, K., Zisserman, A., et al.: Spatial transformer networks. In: Proc. Adv. Neural Inf. Process. Syst. (2015)
5. Karatzas, D., Gomez-Bigorda, L., Nicolaou, A., Ghosh, S., Bagdanov, A., Iwamura, M., Matas, J., Neumann, L., Chandrasekhar, V.R., Lu, S., et al.: Icdar 2015 competition on robust reading. In: Proc. IEEE Int. Conf. Doc. Anal. and Recogn. (2015)
6. Kim, J., Lee, J.K., Lee, K.M.: Accurate image super-resolution using very deep convolutional networks. In: CVPR (2016)
7. Lai, W., Huang, J., Ahuja, N., Yang, M.: Deep laplacian pyramid networks for fast and accurate super-resolution. In: CVPR (2017)
8. Leal-Taixé, L., Roth, S. (eds.): Computer Vision - ECCV 2018 Workshops - Munich, Germany, September 8-14, 2018, Proceedings, Part V. Lecture Notes in Computer Science (2019)
9. Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., et al.: Photo-realistic single image super-resolution using a generative adversarial network. In: Proc. IEEE Conf. Comp. Vis. Patt. Recogn. (2017)
10. Lim, B., Son, S., Kim, H., Nah, S., Lee, K.M.: Enhanced deep residual networks for single image super-resolution. In: CVPR (2017)
11. Luo, C., Jin, L., Sun, Z.: Moran: A multi-object rectified attention network for scene text recognition. Pattern Recognition (2019)
12. Mechrez, R., Talmi, I., Zelnik-Manor, L.: The contextual loss for image transformation with non-aligned data. In: ECCV (2018)
13. Shi, B., Bai, X., Yao, C.: An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. IEEE Trans. Pattern Anal. Mach. Intell. (2017)
14. Shi, B., Yang, M., Wang, X., Lyu, P., Yao, C., Bai, X.: Aster: An attentional scene text recognizer with flexible rectification. IEEE Trans. Pattern Anal. Mach. Intell. (2018)
15. Zhang, X., Chen, Q., Ng, R., Koltun, V.: Zoom to learn, learn to zoom. In: CVPR (2019)
16. Zhang, Y., Tian, Y., Kong, Y., Zhong, B., Fu, Y.: Residual dense network for image super-resolution. In: CVPR (2018)