Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 193 (2021) 190–199

Procedia Computer Science

www.elsevier.com/locate/procedia

10th International Young Scientists Conference on Computational Science

On realistic generation of new format license plate on vehicle images

Anton Maltsev[a], Roman Lebedev[a], Natalia Khanzhina[a,*]

[a]*Machine Learning Lab, ITMO University, 49 Kronverksky Pr., St. Petersburg 197101, Russia*

Abstract

Modern systems of automatic license plate recognition are mostly based on deep neural networks and trained on image data. If data distribution changes, the systems need re-training. For example, new license plate formats result in changing the distribution. Deep neural networks are known to require a huge amount of data for training. The introduction of a new plate shape and sequence of numbers and letters requires the collecting a new training sample, which is impossible due to the lack of the necessary amount of real world examples. The solution is to generate images of the new license plate format based on the old format images while maintaining photorealism. This allows to effectively train systems for both detection and recognition of new license plates and adapt them to real world data in advance. The paper presents a fully automatic approach of photorealistic generation for the new format of Russian license plates. The approach is a sequential algorithm based on deep neural networks, computer vision, projective geometry, and style transfer techniques. It has been tested for the new format of Russian vehicles license plates. The license plates generated by our approach are detected and recognized with better performance as the corresponding old license plates. The approach shows the generalization to the real world data.

Keywords: Image generation; license plate recognition; Optimal Transport; CycleGAN; Neural style transfer; style transfer; Mask R-CNN.

1. Introduction

Since August 2020, a new format of license plates (LP) has been approved in the Russian Federation for installation on vehicles with non-standard mounting points. The new format includes placing the license plate number in two lines instead of one. This implies the need to re-train existing automatic license plate recognition (ALPR) systems. Such systems are usually based on deep learning, at the stage of both LP detection and recognition [1, 2, 3]. To train them for the new LP format, a large number of images is required, which is impossible to provide at the moment due to the low prevalence of the new format vehicles in real world. At the same time, the recognition of new LP is critical, since such vehicles already travel on roads in Russia and other CIS countries. This justifies the need for artificial generation of LP images in a new format to adapt existing ALPR systems to them. Therefore, we have to solve two main problems: the

* Corresponding author. Tel.: +7-953-155-5892.
E-mail address: nehanzhina@itmo.ru

1877-0509
10.1016/j.procs.2021.10.019

photorealistic generation of a new LP images for training recognition models, as well as its photorealistic embedding into real world vehicle images for training detection models. Photorealism is an important property for approximating the distribution of real data. Its compliance during the generation allows to increase the efficiency of solving these problems [15, 16].

The problem of LP generation embedded into real world vehicle images is poorly researched. Thus, works [9, 10, 11, 12], consider only the generation of the LP itself without further embedding. Researchers [9, 10] apply generative adversarial network (GAN) architectures such as CycleWGAN, CycleWGAN-GP to generate Chinese LPs for this purpose. In work [11] the Pix2Pix, CycleGAN, and StarGAN architectures are used to generate Korean LPs. Researchers [12] generate Chinese LPs using WGAN, DCGAN, CycleGAN with a small surrounding context of the vehicle.

The closest to our work is the study [13, 14], in which the Brazilian LPs are converted to the Mercosur format with further embedding in the original image instead of the old license plate. They used the following algorithm. The number plate is detected on the vehicle image using Tiny-YOLOv3 [4]. A new LP is automatically generated. Next, simple computer vision techniques are applied to LP to provide the shadow effect. The resulting LP is turned at a pre-calculated angle and embedded into the place of the old LP.

However, such a simple affine transformation for the LP embedding as well as non-adaptive shadow effects are disadvantages of this approach as they allow no generating photorealistic images. To fill this gap, we apply the projective transformation and style transfer from the original LP to the new LP. In this study we evaluate and choose the best style transfer technique among generative adversarial networks (namely CycleGAN [5]), Optimal Transport [18, 19] and "Neural style transfer" [17] techniques. In addition, our approach includes a license plate number recognition step, which allows to generate images of a vehicle with a new format LP from the image with the old one in an end-to-end manner.

The research questions are the following:

RQ1 Can photorealistic new format license plates be synthetically generated from other LP format images without training data of the new format?

RQ2 Can this synthetic data be effectively used for training license plate detection and recognition models?

The contribution of the work is the proposed end-to-end approach for generating images of vehicles with Russian LPs of the new format from images of the same vehicles with LPs of the old format. This algorithm allows to transfer the effects of lighting, dirt and noise from the original LP and realistically embed the generated LP into the original vehicle image. The code of the proposed approach can be found in `https://github.com/ToxinG/CarLicensePlatesGenerator`.

2. Background on image style transfer

2.1. CycleGAN

As the baseline style transfer technique we chose CycleGAN [5], also used in [9, 10, 11, 12] for LP generation. CycleGAN is trained on two image classes to map images from one class to another. It learns a generalized style of training images, therefore, a specific image cannot be used to transfer style from it. Thus, this architecture does not allow to solve our problem properly, and we use it only to compare results of different style transfer techniques.

2.2. Optimal Transport

The theory of Optimal Transport (OT) [18, 19] is also used for image generation via style transfer. OT technique seeks a generative mapping $g : \mathcal{X} \mapsto \mathcal{Y}$, such that the Transport cost $c : \mathcal{X} \times \mathcal{Y} \mapsto \mathbb{R}$ is minimal:

$$\mathrm{Cost}(\mathbb{P}, \mathbb{Q}) = \min_{g \circ \mathbb{P} = \mathbb{Q}} \int_{\mathcal{X}} c(x, g(x))\, d\mathbb{P}(x), \tag{1}$$

where $\mathbb{P} \in \mathcal{X}$ and $\mathbb{Q} \in \mathcal{Y}$ are input and target image distributions, respectively. This equation is called Monge's formulation of Optimal Transportation.

The primary OT generative technique is based on optimizing the regularized dual form of the Transport cost. It fits two potentials (primal and conjugate) and then uses the barycentric projection to establish the desired generative function. This technique is not end-to-end, as it contains two sequential steps, although it uses a non-minimax optimization objective. Assume $\mathbb{P}$ is the input image and $\mathbb{Q}$ is the target image. Then, OT technique can be used to find the mix of $\mathbb{P}$ with $\mathbb{Q}$ using g generative function in such a way that the new image preserves the content of $\mathbb{P}$ (i.e. contours) but replace its style with $\mathbb{Q}$ (i.e. the color scheme).

2.3. Neural style transfer

This technique is described in [17]. It aims to generate an image x given a content image p and a style image a. It is based on a convolutional neural network (CNN) architecture VGG-19 [20] consisting of 16 convolutional and 5 pooling layers. The max-pooling operation is replaced by average pooling to improve image synthesis results. Each layer of the network is considered here as a non-linear filter bank. Thus, layer l with N_l different filters has N_l feature maps of size M_l, which is the height times the width of the feature map. The feature maps are the responses of a layer l that stored in a matrix $F^l \in \mathbb{R}^{N_l \times M_l}$, where F^l_{ij} is the activation of the filter i at position j of layer l. Let P^l and F^l be the feature representations of the content image p and the generated image x that are provided by the layer l, respectively. The loss function between them is then defined as the squared error:

$$L_{content}(p, x, l) = \frac{1}{2} \sum_{i,j} (F^l_{ij} - P^l_{ij})^2. \tag{2}$$

For the image x initialized with white noise the gradient is computed and x is changed during training by the backpropagation until it generates the same response of a certain CNN layer as the content image p.

At the same time, on the top of each CNN layer responses a style representation is built. This representation is used to match the style of the generated image x and the style image a and presented by the Gram matrix $G^l \in \mathbb{R}^{N_l \times N_l}$, where G^l_{ij} is the inner product of intermediate vectorized feature maps i and j of the layer l. The style matching is performed by minimizing mean-squared distance between the the Gram matrix A^l of the style image a and the Gram matrix G^l of the generated image x. The total style loss is

$$\begin{aligned} L_{style}(a, x) &= \sum_{l=0}^{L} w_l E_l, \\ E_l &= \frac{1}{4N_l^2 M_l^2} \sum_{i,j} (G^l_{ij} - A^l_{ij})^2, \end{aligned} \tag{3}$$

where E_l is the l layer loss, w_l are weighting factors of the contribution of each layer l to the total style loss.

The content loss between x and p VGG representations and the style loss between x and a representations on different VGG layers are jointly minimized to make the image x mix the content of image p with the style of image a. The joint loss function to be minimized is

$$L_{total}(p, a, x) = \alpha L_{content}(p, x) + \beta L_{style}(a, x), \tag{4}$$

where α and β are the weighting factors and the ratio α/β is either 1×10^{-3} or 1×10^{-4}.

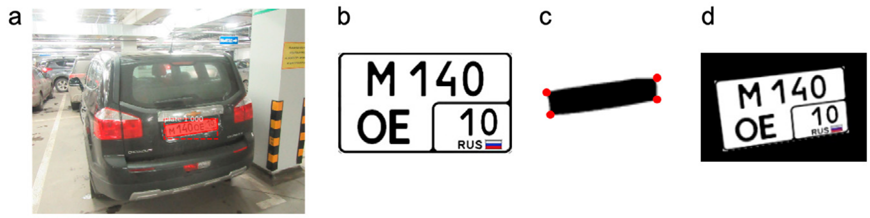

Fig. 1. (a) old LP detection using Mask R-CNN; (b) coarse new LP synthesizing; (c) calculation of detected old LP mask corners; (d) application of perspective transform to new LP.

3. Our approach

To address the problem of the approach [13], in this study we propose the following pipeline for the LP images dataset generation:

- First, the detection of the target (old) license plate on an input image using the neural network detector and cropping detected LPs (see Sec. 3.1);
- Second, the recognition of characters sequence on the target license plate (see Sec. 3.2);
- Third, synthesizing of the coarse new format license plate (see Sec. 3.3);
- Then, the old license plate removal (see Sec. 3.4);
- Next, style transfer from the target license plate to the coarse new LP (see Sec. 3.5);
- Finally, the new license plate embedding in the original vehicle image (see Sec. 3.6).

3.1. Detection

In this work, we assume that there are only one or few vehicles in the camera region of an interest (RoI) and a target license plate is encountered in different rotations and distortions. Based on these assumptions for LP detection we use *instance segmentation* approach with an ability of optional postprocessing stage for license plate coordinates regression. We apply Mask R-CNN [6] architecture with pre-trained ResNet101 [21] backbone to predict a segmentation mask for the license plate. Then, the predicted regions are filtered by their area sizes. Bounding boxes of the remaining ones are cropped. The result of this step is presented in Fig. 1a, c. Different advance techniques to get axis-aligned and minimally distorted license plate can be used here but those are out of the scope of this work.

3.2. Recognition

In general, license plate recognition can be defined as an optical character recognition (OCR) problem. Some OCR approaches use a combination of two models: for character detection to localize license plate characters and for character classification of a predefined alphabet [22, 23]. Most modern approaches [24, 25, 26] are based on Connectionist temporal classification (CTC) loss [7]. These methods are able to predict a LP number characters sequence of arbitrary length from an input image in an end-to-end manner. Output of a model trained with CTC loss has to be processed with some decoding strategies, for example, the greedy search or beam search.

In our study we propose a more simple but effective model that can be trained in an end-to-end manner. As long as the format of a target license plate is fixed, i.e. length is known and each position of the sequence has known domain (letter or digit), the task can be defined as multi-task classification problem. Each NN model head solves for its own task, i.e. prediction of the certain position LP character. As a result, the probability of each character can be obtained based on softmax activation.

Therefore, the model has the following parts: a convolutional feature extractor, a temporal decoder and multiple heads. Semantic features, extracted by pretrained ResNext101-32d [27], are fed to the decoder with bidirectional LSTM [8] layer to address long range connections between characters and then classified by each positional head to

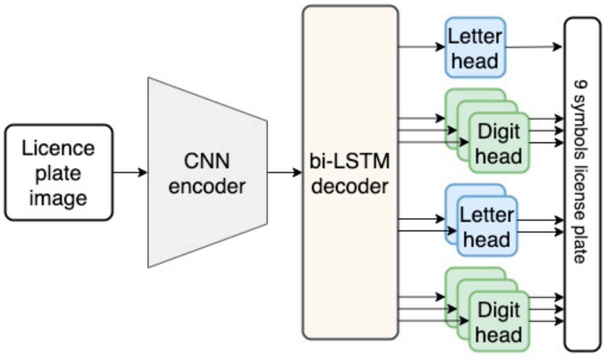

Fig. 2. License plate recognition network architecture. The NN has 9 heads corresponding to the length of Russian LP format.

the target character. High level scheme of the proposed recognition NN for the fixed Russian LP format is shown in Fig. 2.

3.3. Synthesizing of the coarse new format license plate

A coarse image is generated with an algorithm based on functions of `OpenCV` library [28].

First, a schematic image of a resulting new LP is created from the font and template of Russian new format LPs according to previously recognized character sequence. The resulting image is presented in Fig. 1b.

Then, a convex hull is built for the mask obtained by semantic segmentation. The corner points with their surrounding areas are excluded from its contours to leave four nearly straight fragments. We build approximating straight lines for them and the intersections of adjacent ones. These intersections are considered as the corners of the minimal bounding rectangle of the old LP. Using them and LP dimensions we compute matrix M of the perspective transform that maps normalized LP to the one we have in the image. M is computed by the function `getPerspectiveTransform()` of `OpenCV`, which requires four pairs of the corresponding points (LP corners in our case) and solves the equation $\begin{bmatrix} t_i x'_i \\ t_i y'_i \\ t_i \end{bmatrix} = \mathtt{M} \cdot \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix}$ with $dst(i) = (x'_i, y'_i)$, $src(i) = (x_i, y_i)$, $i = 0, 1, 2, 3$, src the coordinates of quadrangle vertices in the source image and dst the coordinates of the corresponding quadrangle vertices in the destination image using Gaussian elimination with the optimal pivot element chosen. This perspective transform is used to turn a new LP in space to fit the image angle. It is applied by the function `warpPerspective()`:

$$\mathtt{dst}(x, y) = \mathtt{src}\left(\frac{M_{11}x + M_{12}y + M_{13}}{M_{31}x + M_{32}y + M_{33}}, \frac{M_{21}x + M_{22}y + M_{23}}{M_{31}x + M_{32}y + M_{33}}\right), \tag{5}$$

where `src` is the input image and `dst` is the output image.

3.4. Old license plate removal

Since the old LPs are significantly wider than the new ones, we need to remove the LP from the image and make its area complementary with its surroundings. To achieve this, we propose the following algorithm.

We apply the perspective transform with the inverse M matrix to the original image to make the LP axis-aligned. The LP from the transformed image is cut off and saved for further use. Then, we observe surrounding areas of the LP to find the small areas with the minimum mean-squared horizontal gradients, which helps avoid non-horizontal

Fig. 3. Results of old license plate removal step on different images

contours. These found small pieces are then used to fill the LP area, alternating with their horizontal reflections to avoid vertical joints.

This method allows to extend vehicle body quite realistically, replacing the removed LP. The resulting images obtained with this method are presented in Fig. 3.

3.5. Style transfer

Image p of the old LP that we cut off in the previous section is resized to a 256×256 square image p_{sq}. After that we apply dilation to it to make characters less standing out and get image $p_{dilated}$. We also resize the coarse image of the new LP to the c_{sq} image with same dimensions as p_{sq}. Then, the style transfer technique is applied to transfer style from p_{sq} and $p_{dilated}$ to c_{sq} to get r_{sq} and $r_{dilated}$ respectively. The pixels of r_{sq} corresponding to the white LP background are replaced with pixels of $r_{dilated}$ in same positions to get r_{final}. This replacement helps avoid artifacts on the new LP background, which may appear because characters of the old LP are treated by the style transferring network as a style. In this study we examine three different image style transfer techniques, presented in section 2.

3.6. New license plate embedding

The LP r_{final} is resized back to the real new format LP dimensions and placed on the transformed image at the position of removed old LP with vertical alignment by the top edges. The perspective transform with matrix M is then applied to obtain the final image. The perspective transform result before embedding is presented in Fig. 1d.

4. Experiments and Results

4.1. Datasets

We used two different datasets to evaluate our approach. The first dataset was used to train style transfer techniques and evaluate their quality. The dataset is scraped from `http://avto-nomer.ru/` and contains 5486 images from various views with width and height varying from 640 to 2048 and from 485 to 1985 pixels, respectively. Images are mostly made at day time and have a target vehicle on a foreground. Fig. 1,3 present processed images from this dataset.

The second dataset was used to test the generalization ability of the recognition and detection models trained on the generated dataset of LPs embedded in vehicle images obtained by the best technique. This is a private dataset of 20 real world images with vehicles having new format LPs.

4.2. Experimental design

To evaluate our approach the following comparisons were performed. First, to choose the best style transfer technique we trained several models on the first dataset. We compared them using the structural similarity index measure (SSIM) [29]. The SSIM is basically a method for evaluation the quality of digital images by comparing their similarity to a reference image. The SSIM is used here as the main generation quality assessment metric in this study and reflects the photorealism of the generated images.

Next, using the the best style transfer technique we applied our approach pipeline to generate images of vehicles with the new format LP. Then, we computed the detection and recognition accuracy on the original dataset with old LPs and generated dataset with new LPs. The comparison is aligned via using the same NN models. Thus, we trained two recognition and two detection models of the same architectures and compared their performance based on the Hamming distance and average precision, respectively. These metrics can be considered as the proxy metrics of the generation quality as they do not assess the photorealism property directly.

Some approaches use the Levenshtein distance (minimal number of insertions, deletions or substitutions required to change one word to another) to measure recognition performance [31, 32, 33]. In our study, as LPs format is fixed, there is no need in additional alignment between the ground truth and the predicted LPs. Hence, we need to count only the number of character substitutions between the ground truth and prediction. As a result, we define the recognition accuracy as a fraction of license plates having the Hamming distance less or equal to some threshold. The Hamming distance of zero value means a perfect match and is mostly used in our comparisons. However, the Hamming distance equal to one is also acceptable for real world systems.

For detection evaluation the standard average precision measure is used. This is the averaged precision obtained with different Intersection over Union thresholds.

Finally, we additionally evaluated the models trained on generated by our approach images on the test dataset of real world images with new format LPs.

For license plate detection and recognition the first dataset train/test split was 4986/500, respectively. Recognition model was trained during 300 epochs with a batch size of 32. During the training phase standard augmentation techniques were used, such as horizontal flip, shifts, scales and rotations, addition of Gaussian noise. Adam optimizer with initial learning rate of 10^{-4} and cosine annealing with warm restarts scheduler were used. The experiments were conducted on a computer with a GeForce GTX 1080 Ti GPU with 128 GB of RAM. The training of the detection and recognition models took about 3 hours.

4.3. Quantitative results

To compare the different techniques of style transfer, namely CycleGAN, Optimal Transport, and Neural style transfer, described in the section 2, the SSIM was calculated for the old-new pairs of LP image. The following algorithm was used to perform it. First, the old license plate was extracted from the original image and transformed to the axis-aligned view. A style transfer technique was then applied to the coarse generated new license plate. Next, the SSIM was calculated for the resulting image pairs, which was then averaged for the whole dataset. The results of the average SSIM are presented in Table 1.

As we can see, all the techniques provided the low SSIM, which can be caused by the difference in the format of LP numbers (one lined versus two lined). As Table 1 shows, the best technique by the SSIM is Neural style transfer and it has an acceptable image generation time. Optimal Transport shows worse result in terms of the SSIM and the worst generation time. Finally, the least realistic was the generation using CycleGAN, although it has the minimal generation time per image. Noteworthy, CycleGAN was used by other researchers and therefore was chosen as a baseline in our work. Thus, we can conclude that the Neural style transfer technique has surpassed the baseline by the main metric. For this reason, it was chosen for further use in our LP generation pipeline. The visual results of the license plate generation by all three techniques under study can be seen in section 4.4.

Table 1. Comparison of style transfer techniques realism property based on SSIM and generation time per image.

Technique	SSIM	Time, s
CycleGAN	0.22	**0.14**
Optimal Transport	0.29	15
Neural style transfer	**0.32**	0.43

The detection and recognition models used for accuracy evaluation are described in sections 3.1, 3.2. The models comparison for old (one line) format and new generated (two lines) LP are presented in Table 2. The detection result is presented on the test set only.

Table 2. Comparison of recognition and detection models performance for old one line LP format and generated two lines LP format.

Data	Recognition train accuracy,%	Recognition test accuracy,%	Detection average precision,%
Old one line LP	95.32	100.0	83.2
New generated two lines LP (our)	**97.53**	100.0	**95.0**

On the test dataset of real world images the recognition model trained on generated license plates of new format predicted 12 license plates correctly and 6 with one error out of total 20 images, while the same model trained only on the old LPs dataset performed significantly worse having the minimum Hamming distance equal to 6. The detection model trained on the new LPs data also significantly outperformed the model trained on the old LPs. The results are presented in Table 3. Thus, the models pre-trained with the synthetic data show acceptable performance especially considering complete absence of real world data for training.

Table 3. Recognition and detection models performance on the test dataset of real world images. Column “# of errors” is the acceptable number of erroneously recognized characters, “Accuracy” is the model accuracy with the acceptable level of errors. Rows for # of errors equal to 4 and 5 are omitted for brevity.

Recognition, # of acceptable errors	New LP model, accuracy,%	Old LP model, accuracy,%
0	60.0	0
1	90.0	0
2	95.0	0
3	100.0	0
6	100.0	20.0
7	100.0	25.0
8	100.0	50.0
9	100.0	100.0
Detection average precision,%	78.1	64.9

4.4. Qualitative results

The images of different style transfer techniques evaluated in this research are presented in Fig. 4. The images obtained with CycleGAN technique (Fig. 4b) are less photorealistic, as the transferred style is rather generic than specific to some particular image. The cause is that CycleGAN learned the most general light characteristics only. Therefore, this architecture is not applicable to transfer style of a particular old LP format vehicle image.

a b c d

Fig. 4. (a) old LPs on original images; new LPs before perspective transform with styles transferred by (b) CycleGAN, (c) Optimal Transport; (d) Neural style transfer.

Next, Optimal Transport technique (Fig. 4c) provides more realistic images. However, the OT seems to build the mapping of pixels hue ignoring the brightness. This results in more uniform color transfer but does not consider the spatial light effects.

Finally, Neural style transfer produced the most realistic results (Fig. 4d). The technique considers the light effects and transfer them preserving spatial position. However, these lights effects seem to be transferred on characters, not only on the background as expected. Also, produced LP images lack motion effects. These issues require further developing. The resulting images of the full pipeline of our approach are presented in Fig. 5.

Fig. 5. Original images (top row) and resulting images obtained by our algorithm (bottom row).

5. Conclusion and discussion

In this paper, we propose a fully automated approach for the synthetic image generation of new format license plates. The approach allows to embed the generated LPs in the old format LP vehicle images. The synthetic dataset generated by the proposed approach may be used to train LP detection and recognition models to be embedded in ALPR systems.

Our experiments showed that both recognition and detection models trained on the dataset generated by our approach improved the accuracy level compared to the old format LPs dataset. Besides, the models trained on our data are well generalized to the real world images of new format LPs. This answers the *RQ2*. If a sufficient number of real world images with the new format LPs appears, the algorithm is useful for data augmentation. In this case the generation quality can be assessed via Fréchet Inception Distance, for example.

A current disadvantage of the approach is that the SSIM is not as high as expected, although the visual analysis of Neural style transfer shows that the images are realistic enough. This issue should be addressed in the future work. The light effects erroneously transferred on the characters also remain for the future research.

The proposed approach is fully automatic, quite realistic and fast, which answers the *RQ1*. In the future, the approach methodology can be modified to generate license plates of other countries, which have limited open datasets.

Acknowledgements

This work is financially supported by National Center for Cognitive Research of ITMO University.

References

[1] Montazzolli, Sérgio and Claudio Jung. (2017) “Real-time brazilian license plate detection and recognition using deep convolutional neural networks”. *30th SIBGRAPI conference on graphics, patterns and images (SIBGRAPI)*: 55-62.

[2] Silva, Sergio Montazzolli and Claudio Rosito Jung. (2018) "License Plate Detection and Recognition in Unconstrained Scenarios". *Proceedings of the European Conference on Computer Vision (ECCV)*.

[3] Selmi, Zied, Mohamed Ben Halima, Umapada Pal and M. Adel Alimi (2020) "DELP-DAR system for license plate detection and recognition". *Pattern Recognition Letters* **129**: 213-223.

[4] Redmon, Joseph and Ali Farhadi. (2018) "Yolov3: An incremental improvement." *arXiv preprint arXiv*:1804.02767

[5] Zhu, Jun-Yan, Taesung Park, Phillip Isola, and Alexei A. Efros (2017) "Unpaired image-to-image translation using cycle-consistent adversarial networks". *Proceedings of the IEEE international conference on computer vision*: 2223-2232.

[6] He, Kaiming, Georgia Gkioxari, Piotr Dollár and Ross Girshick. (2017) "Mask r-cnn". *Proceedings of the IEEE international conference on computer vision*: 2961-2969.

[7] Graves, Alex, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. (2006) "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks". *Proceedings of the 23rd international conference on Machine learning*: 369-376.

[8] Hochreiter, Sepp and Schmidhuber, Jürgen. (1997) "Long short-term memory". *Neural computation*, **9** (8): 1735-1780.

[9] Sun, Ming, et al. (2019) "Image Generation Framework for Unbalanced License Plate Data Set". *In 2019 International Conference on Data Mining Workshops*, IEEE: 883-889.

[10] Wang, Xinlong, et al. (2017) "Adversarial generation of training examples: applications to moving vehicle license plate recognition." *arXiv preprint arXiv*:1707.03124.

[11] Han, Byung-Gil, et al. (2020) "License Plate Image Generation using Generative Adversarial Networks for End-To-End License Plate Character Recognition from a Small Set of Real Images." *Applied Sciences* **10** (8): 2780.

[12] Huang, Jintao, et al. (2019) "Research on vehicle license plate data generation in complex environment based on generative adversarial network." *Eleventh International Conference on Digital Image Processing (ICDIP 2019)*. International Society for Optics and Photonics, **11179**: 1117932.

[13] Silvano, Gilles, et al. (2020) "Synthetic image generation for training deep learning-based automated license plate recognition systems on the Brazilian Mercosur standard." *Design Automation for Embedded Systems*: 1-21.

[14] Silvano, Gilles Velleneuve Trindade, et al. (2020) "Artificial Mercosur license plates dataset." *Data in Brief* **33**: 106554.

[15] Efimova, Valeria, Viacheslav Shalamov, and Andrey Filchenkov (2020) "Synthetic dataset generation for text recognition with generative adversarial networks." *Twelfth International Conference on Machine Vision (ICMV 2019)* **11433**: 1143315.

[16] Khanzhina, Natalia, Natalia Slepkova, and Andrey Filchenkov (2020 "Synthetic images generation for text detection and recognition in the wild." *Twelfth International Conference on Machine Vision (ICMV 2019)* **11433**: 1143312.

[17] Gatys, Leon, Alexander Ecker, and Matthias Bethge. (2016) "A Neural Algorithm of Artistic Style." *Journal of Vision* **16** (12): 326-326.

[18] Villani, C'edric (2008) "Optimal transport: old and new" *Springer Science & Business Media***338**.

[19] Peyr'e, Gabriel, and Marco Cuturi (2019) "Computational optimal transport: With applications to data science" *Foundations and Trends® in Machine Learning* **11** (5-6): 355-607.

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

[21] He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun (2016) "Deep residual learning for image recognition" *Proceedings of the IEEE conference on computer vision and pattern recognition*: 770–778.

[22] Bartz, Christian, Haojin Yang, and Christoph Meinel (2017) "STN-OCR: A single neural network for text detection and text recognition" *arXiv preprint arXiv:1707.08831*.

[23] Liao, Minghui, Jian Zhang, Zhaoyi Wan, Fengming Xie, Jiajun Liang, Pengyuan Lyu, Cong Yao, and Xiang Bai (2019) "Scene text recognition from two-dimensional perspective" *Proceedings of the AAAI Conference on Artificial Intelligence* **3** (1): 8714–8721.

[24] Liu, Xuebo, Ding Liang, Shi Yan, Dagui Chen, Yu Qiao, and Junjie Yan (2018) "Fots: Fast oriented text spotting with a unified network" *Proceedings of the IEEE conference on computer vision and pattern recognition*: 5676–5685.

[25] Busta, Michal, Lukas Neumann, and Jiri Matas (2017) "Deep textspotter: An end-to-end trainable scene text localization and recognition framework" *Proceedings of the IEEE international conference on computer vision*: 2204–2212.

[26] Bušta, Michal, and Patel, Yash, and Matas, Jiri (2018) "E2e-mlt-an unconstrained end-to-end method for multi-language scene text" *Asian Conference on Computer Vision, Springer*: 127–143.

[27] Xie, Saining, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He (2017) "Aggregated residual transformations for deep neural networks" *Proceedings of the IEEE conference on computer vision and pattern recognition*: 1492–1500.

[28] Bradski, G. (2000) "The OpenCV Library" *Dr. Dobb's Journal of Software Tools*.

[29] Wang, Zhou, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli (2004) "Image quality assessment: from error visibility to structural similarity" *IEEE transactions on image processing* **13** (4): 600–612.

[30] Levenshtein, Vladimir I et al. (1966) "Binary codes capable of correcting deletions, insertions, and reversals" *Soviet physics doklady* **10** (8): 707–710.

[31] Adhitama, Perdana, Soo Hyung Kim and In Seop Na (2014) "Lexicon-Driven Word Recognition Based on Levenshtein Distance" *International Journal of Software Engineering and Its Applications* **8** (2): 11–20.

[32] Khan, Shamaila and Ali Sarfraj (2016) "Vehicle Plate Matching using License Plate Recognition based on Modified Levenshtein Edit Distance" *International Journal of Computer Applications* **975**: 8887.

[33] Coustaty, Mickaël, Antoine Doucet, Adam Jatowt and Nhu-Van Nguyen et al. (2018) "Adaptive edit-distance and regression approach for post-OCR text correction" *International Conference on Asian Digital Libraries*: 278–289.