sensors

Communication

Unsupervised Domain Adaptive Corner Detection in Vehicle Plate Images

Kyungkoo Jun

Department of Embedded Systems Engineering, Incheon National University, Incheon 22012, Korea; kjun@inu.ac.kr

Abstract: Rectification of vehicle plate images helps to improve the accuracy of license-plate recognition (LPR). It is a perspective-transformation process to project images as if taken from the front geometrically. To obtain the projection matrix, we require the (x, y) coordinates of four corner positions of plates in images. In this paper, we consider the problem of unsupervised domain adaptation for corner detection in plate images. We trained a model with plate images of one country, the *source domain*, and applied a domain adaptation scheme so that the model is able to work well on the plates of a different country, the *target domain*. For this study, we created a dataset of 22,096 Korea plate images with corner labels, which are source domain, and 6762 Philippines, which are target domain. To address this problem, we propose a heatmap-based corner-detection model, which outperforms existing scalar-regression methods, and an image classifier for mixed image of source and target images for domain adaptation. The proposed approach achieves better accuracy, which is 19.1% improvement if compared with baseline discriminator-based domain adaptation scheme.

Keywords: license plate recognition; rectification; corner detection; domain adaptation; heatmap

Citation: Jun, K. Unsupervised Domain Adaptive Corner Detection in Vehicle Plate Images. *Sensors* **2022**, *22*, 6565. https://doi.org/10.3390/s22176565

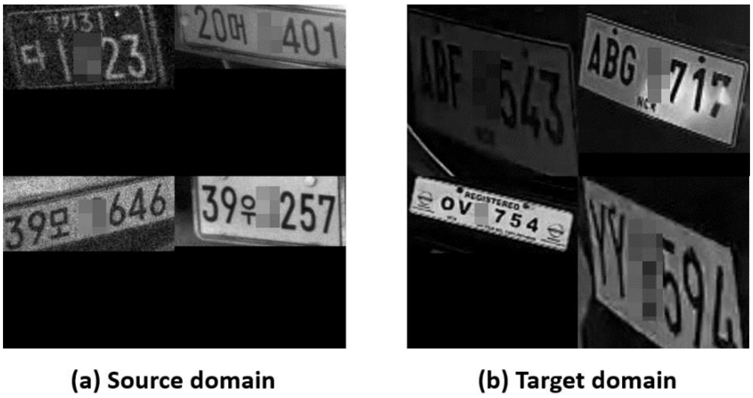
Academic Editor: Giambattista Gruosso

Received: 19 July 2022
Accepted: 29 August 2022
Published: 31 August 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

1. Introduction

License-plate recognition (LPR), in general, consists of two steps [1,2]. In step one, the license plate area is localized, and then in step two, character recognition is performed within the plate area. When the localized plate areas are not properly rectified, as shown in Figure 1, the accuracy of the character recognition degrades. Thus, the warping of plate images as if seen from the straight front would help to improve the performance of LPR. We call this transformation process rectification. Other benefits of the rectification are twofold: (1) it makes the labeling work easier and (2) reduces the dependency on image augmentation during training phase.


(a) Source domain

(b) Target domain

Figure 1. Examples of non-rectified vehicle plates: (**a**) source KR and (**b**) target PH.

The rectification can be easily achieved by perspective transformation [3]. It is to shift one view angle of an object to another view of the same object. This process involves $\mathcal{H}$, a homography 3 × 3 matrix, as shown in Equation (1):

$$\begin{bmatrix} x_i' \\ y_i' \\ 1 \end{bmatrix} = \mathcal{H} \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} x_i' \\ y_i' \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} h_{12} h_{13} \\ h_{21} h_{22} h_{23} \\ h_{31} h_{23} h_{33} \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} \tag{1}$$

where $(x_i, y_i, 1)$ and $\left(x_i',\ y_i', 1\right)$ are homogeneous coordinates of a point before and after perspective transformation, respectively. To determine $\mathcal{H}$, we need four pairs of corresponding points of $(x_i,\ y_i)$ and $\left(x_i',\ y_i'\right)$ for $i = 1 \ldots 4$, which has the following matrix relationship of Equation (2). It represents eight equations to determine the 8 degrees of freedom of $\mathcal{H}$, excluding h_{33}, which is fixed to 1 in homogeneous coordinate system. $\mathcal{H}$ can then be obtained by applying inverse matrix multiplication to Equation (2).

$$\begin{bmatrix} x_1' \\ y_1' \\ x_2' \\ y_2' \\ x_3' \\ y_3' \\ x_4' \\ y_4' \end{bmatrix} = \begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -x_1x_1' & -x_1y_1' \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -x_1y_1' & -y_1y_1' \\ x_2 & y_2 & 1 & 0 & 0 & 0 & -x_2x_2' & -x_2y_2' \\ 0 & 0 & 0 & x_2 & y_2 & 1 & -x_2y_2' & -y_2y_2' \\ x_3 & y_3 & 1 & 0 & 0 & 0 & -x_3x_3' & -x_3y_3' \\ 0 & 0 & 0 & x_3 & y_3 & 1 & -x_3y_3' & -y_3y_3' \\ x_4 & y_4 & 1 & 0 & 0 & 0 & -x_4x_4' & -x_4y_4' \\ 0 & 0 & 0 & x_4 & y_4 & 1 & -x_4y_4' & -y_4y_4' \end{bmatrix} \begin{bmatrix} h_{11} \\ h_{12} \\ h_{13} \\ h_{21} \\ h_{22} \\ h_{23} \\ h_{31} \\ h_{32} \end{bmatrix} \tag{2}$$

For example, we are given image A, which contains a non-rectangular shape of a plate P. Let $\{(x_i, y_i)\}_{i=1}^{4}$ denote the four corner coordinates of P. The corners correspond to left-top, right-top, left-bottom, and right-bottom in order. Then our goal is to generate a new image, B, in which P is transformed into a rectangular-shaped P'. The perspective transformation enables this process if $\mathcal{H}$ is determined. Let $\left\{\left(x_i', y_i'\right)\right\}_{i=1}^{4}$ indicate the four corner positions of P' in B. Note that $\left(x_i',\ y_i'\right)$ is known in advance; they are the predetermined positions where we want to place the transformed plate in B. Therefore, $\{(x_i, y_i)\}_{i=1}^{4}$ in A is the information we need to find in order to calculate $\mathcal{H}$.

This work proposes a novel deep model that locates four corners $\{(x_i, y_i)\}_{i=1}^{4}$ from plate images. Inspired by the latest keypoint detection models, our model is designed to output a set of four heatmaps. The heatmaps correspond to each one of four corners. They embed 2D Gaussian blobs, which represent the probabilities of corner locations. The requirements for the rectification are for real time and have resource efficiency. Considering the entire LPR process, less emphasis is put on the rectification compared with the localization and the recognition. It means that the rectification component should use small and shallow models that consume less computing time and resources.

In this work, another challenge is that we do not assume that training and test data are collected from an identical distribution. The training data are Korean plates, and the test data are Philippines plates. Such a distribution discrepancy has been observed to cause a performance drop in various works. The Korean and Philippines plates are different from each other in regard to shapes, sizes, and characters. Hence, a corner-detection model trained with the Korean plates would not work well with the Philippines. Thus, it is desirable to develop methods that can adapt corner-detection models to unseen domains, which are visually different from the training data. We address this cross-domain corner-detection problem by developing algorithms to transfer knowledge from a labeled source domain of Korean plate images to an unlabeled Philippines.

Domain adaptation techniques [4] aim at reducing the domain gap between the source and target domain. In particular, adversarial discriminative domain adaptation methods [5] have been proposed to improve performance by discouraging discrimination capability between source and target domain data. Our work proposes a different type of data discrimination. We trained the discriminator to tell how source and target domain images are cut and pasted in a new image. This task is more challenging than binary domain discrimination. The discriminator trained with this scheme becomes harder to be deceived; as a result, the feature extractor is guided to capture more domain-invariant plate features.

The contribution of this work can be summarized as follows.

1. We proposed a corner-detection model which outputs a likelihood heatmap. It outperforms existing models based on scalar regression. In addition, we created a new dataset consisting of 22,096 KR plate images and 6762 PH plate images.
2. We presented a novel domain adaptation based on an adversarial discriminative method. It is peculiar in the sense that the discriminator is required to distinguish how Korean plates and target Philippines plates are mixed.
3. We conducted experiments on various corner-detection tasks and validated that our method can bring performance gains by 19.1% if compared with baseline discriminator-based domain adaptation method.

This paper is organized as follows: Section 2 surveys existing works related with our work in regard to subjects such as rectification and domain adaptation, Section 3 presents our proposed model and domain adaptation scheme, Section 4 shows evaluation results, and Section 5 concludes the paper.

2. Related Work

Corner Detection: Finding the four corner positions of vehicle plates for the rectification purpose can be considered as a keypoint detection problem. CNN-based methods [6,7] use features or latent vectors extracted through a sequence of convolution layers to regress corner coordinates. Instead of regressing corners, the methods [8,9] tried to directly generate rectified images. However, these methods are impractical considering the real-time constraint of LPR tasks and degradation of output quality; the image generation models are, in general, computation intensive, and the output images often contain blurred parts.

Recently, Vision Transformer (ViT) [10] is proposed, which depends on a representative transformer model for natural language processing. ViT converts the input image into a sequence of image patches and extracts context information through a self-attention structure with multi-heads. ViT was used to unwarp document images with geometric distortion [11]. The self-attention mechanism, a core element of ViT, is limitedly used only for low-resolution image input due to computational complexity. Moreover, there have been questions about whether it will be effective for tasks that require high input resolution, such as detection or segmentation. For the detection task, a model has been proposed that uses ViT as a backbone [12] with limited success.

Keypoint Detection: 2D keypoint detection is actively studied for its wide applicability in computer vision tasks. Most of the works are based on heatmap-output networks. A multi-resolution framework was proposed that generates heatmaps representing per-pixel likelihood for keypoints [13]. Hourglass networks for heatmaps of human body keypoints were developed for pose estimation [14]. The heatmaps from deconvolutional-layer-added ResNet [15] was utilized for human pose determination.

Domain Adaptation and Keypoint Detection: When domain shift occurs, most deep models experience performance deterioration because of unseen data. One of active works to deal with such degradation is domain adaptation. It has been actively studied in computer vision. Earlier works have mostly focused on image classification. Recent works have widened its application areas, aiming to improve the domain adaptability of deep neural networks, including References [16–20]. We focus on the keypoint detection task, which has been studied lately compared to other areas.

Some previous works have investigated domain adaptation in keypoint detection. Most works are related with 3D keypoints detection. A weakly supervised method using depth images was proposed in Reference [21], domain adaptation with a 3D geometric constraint-aware loss was studied in Reference [22], and prediction regularization for unlabeled target domain in 3D keypoints detection has been enforced by view-consistency during domain adaptation [23]. Our problem setup is different from those works in that we considered 2D image keypoint detection and the domain adaptation.

3. Corner-Detection Model and Domain Adaptation

An overview of our network is shown in Figure 2. It consists of two main parts; the upper part of Figure 2 is the corner-detection components, and the lower part is a classifier, which plays a role as the domain adaptation component in our scheme. The classifier distinguishes input image as one of N label types by giving out one-hot vector of length N. The details about the classifier and its output vector are discussed later.

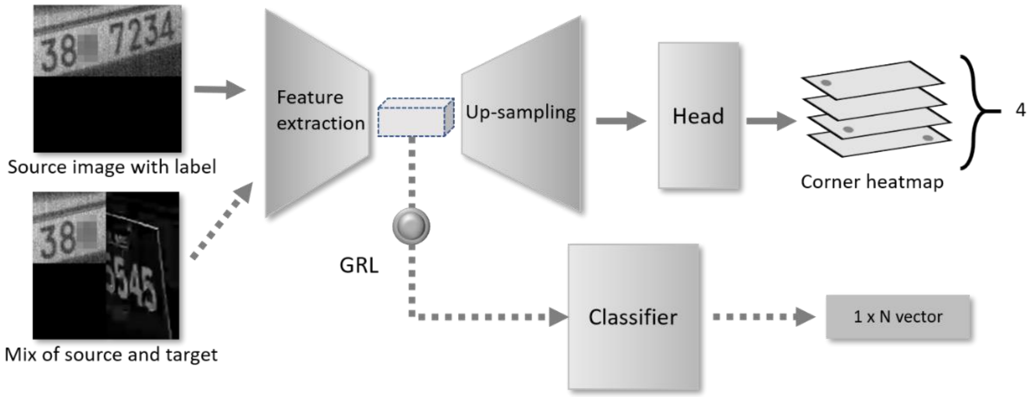


Figure 2. The architecture of the proposed corner-detection model using adversarial mixed image discriminator.

Corner-Detection Model: The corner-detection model extracts a set of likelihood heatmaps about the corner positions from input images. The feature extractor is the backbone of ResNet18 pretrained on ImageNet, followed by the up-sampling component using deconvolution layers. The head predicts four corner locations, namely p_1, p_2, p_3, and p_4 for top-left, top-right, bottom-left and bottom-right, respectively, using heatmap $f(p_k) \in R^{H' \times W'}$ for corner point p_k. The heatmaps contain 2D Gaussian blobs centered at the predicted corner locations. The corner position (x_k, y_k) is determined by the following:

$$J(f\ (p_k)) = arg\ \underset{x,y}{max} f(p_k)_{x,y} \tag{3}$$

Figure 3 shows the corner locations, p_k, of images and corresponding heatmaps. The four heatmaps correspond to each of the corner positions.

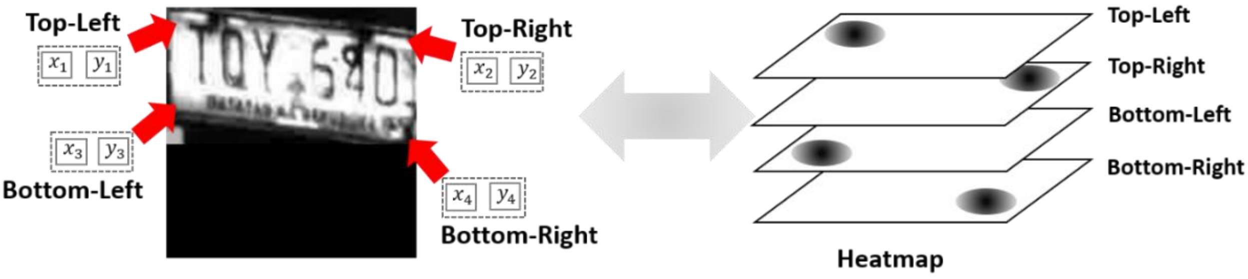


Figure 3. Corner positions and corresponding heatmaps. Each heatmap is responsible for only one corner.

Let us denote $H_i(p_k) \in R^{H' \times W'}$ as the heatmap label of the corner point, p_k, for the i-th training image, with 2D Gaussian blob centered on the ground truth coordinate of p_k. Then we can use the cross-entropy loss, and the heatmap loss can be written as follows:

$$\mathcal{L}_H = -\sum_{i,k}[H_i(p_k) \log f_i(p_k) + (1 - H_i(p_k)) \log(1 - f_i(p_k))] \tag{4}$$

Domain Classifier: The domain classifier in our network is the domain adaptation component to align the feature representation distributions on image level. The image-level representation refers to the feature map from the feature extractor. To reduce the difference between the domain distributions on the image level, the classifier consists of repetitions of a sequence of a convolutional layer, batch normalization, and ReLU activation, followed by an adaptive average pooling and a fully connected layer.

For the classifier training, images with Korean and Philippines plates cut in half and put together side-by-side are used. The classifier predicts the cut-and-paste label for the images. The output dimension of the classifier, $1 \times N$, in Figure 2 indicates the number of classification classes. Let us denote D_i as the label of the i-th training image, $D_i = 0$ for Korean on the left half and Philippines right half, and $D_i = 1$ for the opposite. Therefore, in the current settings, we set $N = 2$. When devising new mix style for classification in the future, we can add new label types by increasing N, which generally helps to improve the adaptation capability of the proposed model.

The benefits of this image mix-up are twofold: (1) the mix-up is generally harder dataset. Thus, the domain adaptation phase leads to more robust results. The performance comparison results when using source and target images separately proves this claim in the following section, and (2) the feature extractor is guided to eliminate the character-specific features in the feature maps, which helps to maximize the classification loss in the adversarial training phase. As a result, the feature maps are more likely to contain more plate-shape-related information.

By denoting the output of the classifier as c_i and using the cross-entropy loss, the domain adaptation loss can be written as follows:

$$\mathcal{L}_{DA} = -\sum_{i}[D_i \log c_i + (1 - D_i) \log(1 - c_i)] \tag{5}$$

To align the domain distributions, we should simultaneously optimize the parameters of the classifier to minimize the $\mathcal{L}_{DA}$ and also optimize the parameters of the feature extractor to maximize $\mathcal{L}_H$. For the implementation, we use the gradient reverse layer (GRL) [15], whereas the ordinary gradient descent is applied for training the classifier. The sign of the gradient is negated when passing through the GRL layer to optimize the feature extractor.

The overall network of Figure 2 is involved only in the training phase. During inference, one can exclude the domain adaptation components, such as the classifier and the GRL, and simply use the corner-detection model with adapted parameters.

4. Experiments

Experiment Setup: The dataset consists of two domains: Korean plate images with a total of 22,096 and Philippines with a total of 6762 images. Following the common terminology in domain adaptation, we refer to the Korean images as source domain, denoted by KR, and to the Philippines images as target domain, PH. All of the images have the same dimension of 416 × 416 and are gray with one channel. For each domain, we split the images into 8:2 ratio for training and test purpose. The corner prediction model outputs are the heatmaps of the dimension of 104 × 104.

We construct the ground truth heatmaps from the labels, which have the normalized coordinates of the four corner positions. The heatmaps have 2D Gaussian blobs centered at the corner locations. We determine the blob size by standard deviation, σ.

We finetune ResNet18 of the feature extractor pretrained on ImageNet. The up-sampler and the heatmap head are trained from scratch, with an initial learning rate of 1×10^{-3}. We adopt mini-batch SGD with a momentum of 0.9 and a batch size of 64. The learning rate is adjusted by $\eta_p = \eta_0\, \alpha^p$, where p is the training steps, $\eta_0 = 1^{-3}$, and $\alpha = 0.95$. Our model is trained for 500 iterations. We chose the optimal model state guided by the highest performance on the validation sets of KR and PH.

Four models are involved in the experiments, as listed in Table 1. The sizes and the parameter numbers of the models are shown. Conv. and MNet. are from the existing works [6,7]. Those models work in a regression way, predicting normalized coordinates of the corners. We do not apply any domain adaptation scheme to those models. We train them by using only the source-domain data. Classic-DA is the corner-detection model with well-known domain adversarial discriminator [24]. The architecture of this model is similar to ours except that the discriminator is used instead of the classifier. Proposed-DA is our model. It reduces the number of parameters slightly compared with Classic-DA because the classifier size is smaller than the discriminator of the Classic-DA.

Table 1. The sizes and number of parameters of the modes used in the experiments.

Model	Model Size (MB)	No. of Parameters (K)
Conv.	352	92,203
MNet.	17	4150
Classic-DA	64	16,563
Proposed-DA	64	15,963

Regarding training data, Conv. and MNet. use only the source images and corresponding labels because the target images have no label information that is required for supervised training. For Classic-DA and Proposed-DA training, both the source and target images are used. Moreover, we train Proposed-DA with only source images and only target images, denoted by Source-Only, and Target-Only, respectively. These are used as baselines to assess the effectiveness of the proposed method.

As evaluation metric, we measure error distances between predicted corner positions and the ground truth are measured in pixel units. The errors for the four corner positions, Left-Top, Right-Top, Left-Bottom, and Right-Bottom, respectively, are measured separately. We run two experiments, one using source KR as the test dataset, and the other with target PH as the test dataset.

Experimental Results: Table 2 summarizes the evaluation results from testing with source KR dataset. Since the test dataset includes no target images, it is not related to domain adaption. Its purpose is to obtain the baseline performance for comparison and evaluate the heatmap regression of the proposed model. The downward arrows in the table mean that the lower the values are, the more accurate the prediction is. We observe that all the heatmap-based methods of Source-Only, Classic-DA, and Proposed-DA show less error than non-heatmap based methods of Conv. and MNet. Thus, note that the heatmap scheme is more effective for regressing the corner locations than scalar value regression.

Table 2. Error distances in pixel for KR source domain images. The least errors are in bold.

Method	Left-Top	Right-Top	Left-Bottom	Right-Bottom	Avg.
Conv.	9.89	10.50	11.65	8.15	10.9
MNet.	6.78	7.19	8.07	5.30	6.8
Source-Only	**5.2**	**5.8**	6.1	**4.4**	**5.4**
Classic-DA	5.74	6.13	**6.08**	4.5	5.6
Proposed-DA	5.69	5.90	6.27	4.66	5.6

Observing the results of the heatmap based Source-Only, Classic-DA, and Proposed-DA, we find that they show similar errors. Recall that the training of Source-Only is different from Classic-DA and Proposed-DA; its training uses only the source KR dataset, while two models use both KR and PH dataset. This finding verifies that the target images included in training have little influence on the proposed domain adaptive model. The model is able to maintain accuracy about the source domain, even though it is trained to prepare for domain adaptation. Interestingly, Source-Only slightly outperforms the other two methods. It is obvious because the models trained and tested with same domain dataset perform best.

The test results for the target PH dataset are summarized in Table 3. In this experiment, we use the model of Target-Only, which is trained and tested only with the target dataset. It is no surprise that Target-Only shows the best results. It is because the model is trained and tested in the same domain. We present its results here just for providing an example of the best accuracy that the domain adaptive schemes should achieve.

Table 3. Error distances in pixel of the models for PH target domain images.

Method	Left-Top	Right-Top	Left-Bottom	Right-Bottom	Avg.
Conv.	25.84	31.69	37.23	32.43	31.8
MNet.	18.57	21.12	19.05	25.34	21.0
Target-Only	6.51	8.09	9.00	8.53	8.03
Source-Only	18.0	20.2	24.1	24.0	21.6
Classic-DA	17.08	20.4	21.2	**20.8**	19.9
Proposed-DA	**10.34**	**13.36**	**15.39**	20.96	**15.01**

These experiment results for the source and target dataset highlight that the domain adaptation methods generalize better than the Source-Only model. The proposed method of Proposed-DA is in general more robust to domain shift than the Classic-DA. This suggests that the classifier for mixed images was more effective for decreasing the domain gap between the two distributions than the image discriminator. It is worth noting that the proposed method exhibits a drastic performance improvement of 19.1% on average if compared with the results of the Class-DA.

Figure 4 shows the qualitative prediction results of the proposed domain adaptive corner detection. We show the ground-truth corners as green circles and the predicted corner locations as the red dots. Some of the characters on the plates are intentionally blurred for privacy protection. In general, the proposed method is able to locate most corners of unseen PH plates. As can be noted, the plate images with clear and distinctive edges show the successful prediction results. Regarding the incorrect results, possible reasons are as follows: (1) plates are partly broken (second row, fourth from left), (2) the boundary lines are blurred with background (second row, first from right), and (3) defects on the plates are present (second row, first from left).

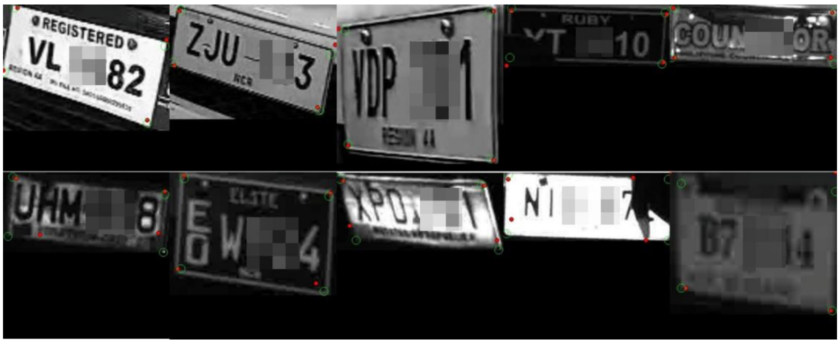


Figure 4. Qualitative evaluation results of the proposed domain adaptation scheme.

Experiments with deeper backbone networks: We also evaluated with deeper backbones such as ResNet34, ResNet50, etc., to assess its effect on the performance. In the current architecture, ResNet18 is adopted as the backbone. As mentioned in Section 1, the rectification component has a limitation in regard to adopting heavier models due to real-time and resource constraint. Since ResNet18 is smaller and faster than other models, it helps LPR to meet real-time requirements. In this experiment, we evaluated the tradeoff between speed and accuracy by employing deeper networks.

For comparison among deeper backbones, we tested with ResNet34, ResNet50, ResNet101, and ResNet152, along with the original backbone of ResNet18. We prepared a set of Proposed-DA models, each with those different backbone networks. The models were trained with the same dataset and parameters until the loss decrease stabilizes. The evaluation used the same test datasets: source KR test dataset and target PH dataset. We used two comparison metrics: the corner errors and the inference time.

As shown in Table 4 for the source KR and Table 5 for the target PH, concerning the prediction errors, the deeper backbones such as ResNet50 and higher models outperform shallow backbones such ResNet18 in both test datasets. However, the error improvements are not substantial, indicating that ResNet18 is sufficiently accurate to be used in real-time environment. Interestingly, the deepest models of ResNet101 and ResNet152 show larger errors than the shallower ResNet50. The training difficulties often encountered with deep and huge models are presumed to cause such lower accuracy.

Table 4. Errors for source KR dataset of Proposed-DA with different backbone networks.

Backbone	Left-Top	Right-Top	Left-Bottom	Right-Bottom	Avg.
ResNet18	5.69	5.90	6.27	4.66	5.6
ResNet34	5.50	5.80	5.97	4.57	5.46
ResNet50	**5.30**	**5.50**	**5.80**	**4.42**	**5.25**
ResNet101	5.53	5.84	6.13	4.70	5.55
ResNet152	5.52	5.90	6.18	4.69	5.57

Table 5. Errors for target PH dataset of Proposed-DA with different backbone networks.

Backbone	Left-Top	Right-Top	Left-Bottom	Right-Bottom	Avg.
ResNet18	10.34	13.36	15.39	20.96	15.01
ResNet34	10.15	12.97	15.22	18.45	14.19
ResNet50	**10.01**	**12.80**	**14.80**	**17.90**	**13.87**
ResNet101	10.04	13.01	15.10	19.03	14.29
ResNet152	10.03	13.04	15.05	19.22	14.33

Table 6 shows average elapsed time (ms) required to predict four corners when given a single image to the Proposed-DA. The images have the dimension of 416 × 416. We measure the times in two separate environments: CPU and GPU. The reason to consider CPU for comparison is that LPR in real-world applications often runs on platforms without GPU. The times are averaged over source KR and target PH test dataset images. As expected, the ResNet18-based model runs fastest, and the deeper models are almost 10 times slower. The results imply that the deeper networks are not suitable for real-time LPR applications, which should finish within, at most, 500 ms, including localization and recognition. In this sense, ResNet18 is a proper selection for the backbone of our proposed model.

Table 6. Average elapsed time (ms) for a single image corner prediction of Proposed-DA.

Backbone	**Core-i7 CPU**	**Nvidia GTX 1660 Super (6G)**
ResNet18	64.5 ms	8.8 ms
ResNet34	109.3 ms	17.4 ms
ResNet50	540.8 ms	52.1 ms
ResNet101	1100.9 ms	98.3 ms
ResNet152	1645.1 ms	120.7 ms

5. Conclusions

We considered the problem of unsupervised domain adaptation for plate corner detection. To conduct our study, we prepared a new dataset consisting of 22,096 KR plate images and 6762 PH plate images, which we can provide under certain agreement. Our main contributions are twofold; we proposed a heatmap-based corner-detection model, which is better than the existing scalar-regression-based methods, and the domain adaptation using mixed image classifier, an effective approach for cross-domain object detection. Our approach is validated on experiments, and our method outperforms the baseline discriminator-based domain adaptation scheme with 19.1% improved accuracy, as well as existing corner-detection methods, thus demonstrating its effective for corner detection in a domain-shift environment.

Although our method can achieve compelling results in the target PH dataset, the results are far from generalization. It may fail when testing with other license plates of different countries. Future work may extend the proposed domain adaptation scheme to become more general, and to other application scenarios beyond license plates and will explore the possibility of combining rotation and flip techniques in adaptation steps to improve performance on the current test dataset. We hope that the proposed dataset will motivate research on this topic and our domain adaptation scheme will serve as a strong baseline for future works.

Funding: This work was supported by an INU research grant of 2018-0393.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Chen, R.C. Automatic License Plate Recognition Via Sliding-Window Darknet-YOLO Deep Learning. *Image Vis. Comput.* **2019**, *87*, 47–56.
2. Menotti, D.; Chiachia, G.; Falcão, A.X.; Neto, V.O. Vehicle License Plate Recognition with Random Convolutional Networks. In Proceedings of the 27th SIBGRAPI Conference on Graphics, Patterns and Images, Rio de Janeiro, Brazil, 26–30 August 2014; Institute of Electrical and Electronics Engineers (IEEE): Piscataway, NJ, USA, 2014; pp. 298–303.
3. Nguyen, T.; Chen, S.W.; Shivakumar, S.S.; Taylor, C.J.; Kumar, V. Unsupervised Deep Homography: A Fast and Robust Homography Estimation Model. *IEEE Robot. Autom. Lett.* **2018**, *3*, 2346–2353. [CrossRef]
4. Wang, M.; Deng, W. Deep Visual Domain Adaptation: A Survey. *Neurocomputing* **2018**, *312*, 135–153. [CrossRef]
5. Tang, H.; Jia, K. Discriminative Adversarial Domain Adaptation. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; pp. 5940–5947.
6. Yoo, H.; Jun, K. Deep Homography for License Plate Detection. *Information* **2020**, *11*, 221. [CrossRef]
7. Yoo, H.; Jun, K. Deep Corner Prediction to Rectify Tilted License Plate Images. *Multimed. Syst.* **2021**, *27*, 779–786. [CrossRef]
8. Lee, Y.; Lee, J.; Ahn, H.; Jeon, M. SNIDER: Single Noisy Image Denoising and Rectification for Improving License Plate Recognition. In Proceedings of the IEEE International Conference on Computer Vision Workshops, Seoul, Korea, 27–28 October 2019.
9. Huang, X.; Huang, Y.; Pei, Y. DocGAN: Document Image Unwarping for High-Level Vision Task. In Proceedings of the IET 8th International Conference on Wireless, Mobile, Multimedia Networks, Beijing, China, 15–17 November 2019.
10. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv* **2020**, arXiv:2010.11929.
11. Feng, H.; Wang, Y.; Zhou, W.; Deng, J.; Li, H. DocTr: Document Image Transformer for Geometric Unwarping and Illumination Correction. *arXiv* **2021**, arXiv:2110.12942.
12. Beal, J.; Kim, E.; Tzeng, E.; Park, D.H.; Zhai, A.; Kislyuk, D. Toward Transformer-Based Object Detection. *arXiv* **2020**, arXiv:2012.09958 2020.

13. Tompson, J.J.; Jain, A.; LeCun, Y.; Bregler, C. Joint Training of a Convolutional Network and a Graphical Model for Human Pose Estimation. In Proceedings of the 27th International Conference on Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014.
14. Newell, A.; Yang, K.; Deng, J. Stacked Hourglass Networks for Human Pose Estimation. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; pp. 483–499.
15. Xiao, B.; Wu, H.; Wei, Y. Simple Baselines for Human Pose Estimation and Tracking. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 466–481.
16. Ganin, Y.; Lempitsky, V. Unsupervised Domain Adaptation by Backpropagation. In Proceedings of the International Conference on Machine Learning, Lille, France, 6–11 July 2015; pp. 1180–1189.
17. Ghifary, M.; Kleijn, W.B.; Zhang, M.; Balduzzi, D.; Li, W. Deep Reconstruction-Classification Networks for Unsupervised Domain Adaptation. In *European Conference on Computer Vision*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 597–613.
18. Sener, O.; Song, H.O.; Saxena, A.; Savarese, S. Learning Transferrable Representations for Unsupervised Domain Adaptation. In Proceedings of the 30th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain, 5–10 December 2016.
19. Panareda Busto, P.; Gall, J. Open Set Domain Adaptation. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 754–763.
20. Motiian, S.; Piccirilli, M.; Adjeroh, D.A.; Doretto, G. Unified Deep Supervised Domain Adaptation and Generalization. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 5715–5725.
21. Cai, Y.; Ge, L.; Cai, J.; Yuan, J. Weakly-Supervised 3d Hand Pose Estimation from Monocular Rgb Images. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 666–682.
22. Zhou, X.; Huang, Q.; Sun, X.; Xue, X.; Wei, Y. Towards 3d Human Pose Estimation in the Wild: A Weakly-Supervised Approach. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 398–407.
23. Zhou, X.; Karpur, A.; Gan, C.; Luo, L.; Huang, Q. Unsupervised Domain Adaptation for 3d Keypoint Estimation Via View Consistency. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 137–153.
24. Tzeng, E.; Hoffman, J.; Saenko, K.; Darrell, T. Adversarial Discriminative Domain Adaptation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7167–7176.