Unsupervised Day-Night Domain Adaptation with a Physics Prior for Image Classification

Gees Brouwer[1]
Supervisor: Attila Lengyel[1] , **Responsible professor: Jan van Gemert**[1]
[1]EEMCS, Delft University of Technology, The Netherlands
{g.d.brouwer}@student.tudelft.nl, {a.lengyel, j.c.vangemert}@tudelft.nl

Abstract

While deep neural networks show great potential for being part of safety-critical applications such as autonomous driving, covering their sensitivity to illumination shifts by adding training data is often non-trivial. The undesired illumination shift between train and test data can be addressed by domain adaptation methods. Recent work [9] has demonstrated performance improvements with a novel zero-shot domain adaptation setting by introducing a physics-based visual inductive prior - a trainable Color Invariant Convolution (CIConv) layer - aiming to transform its input to a more domain invariant representation.
We compare the performance of image classification for day-night domain adaptation in the zero-shot and the unsupervised setting, and explore the effectiveness of using CIConv in both settings. We show that unsupervised domain adaptation reduces the day-night distribution shift similarly to CIConv in the zero-shot setting. We demonstrate improved performance when CIConv and unsupervised day-night domain adaptation are combined.

1 Introduction

Image classification is one of the tasks in which deep neural networks excel. The process of correctly assigning a label to an image can be useful in a lot of applications. For a neural network to be used in safety-critical computer vision applications such as autonomous driving, it is essential to predict labels reliably while dealing with varying recording conditions. Illumination shifts, caused by for example the time of day or weather, are recording conditions deep image recognition methods are sensitive to [1; 2; 15]. Adding extra data for training a model to sufficiently improve its robustness to illumination shifts is often non-trivial as it may be expensive, time-consuming, or even impossible to obtain samples for all possible scenarios in advance.

Instead of adding more data, a visual inductive bias can be used. A great example of such a bias is the Convolutional Neural Network (CNN), using the convolution operator for translation invariance. This saves a massive amount of training data as the network no longer requires samples in different locations.

An illumination shift between train and test data can be addressed by domain adaptation methods [13; 14]. Recent work [9] introduced a novel zero-shot domain adaptation method for addressing day-night domain shifts by exploiting learnable photometric invariant features as a physics-based visual inductive prior; Color Invariant Convolution (CIConv). CIConv is a learnable color invariant CNN layer that aims to transform the input to a domain invariant representation. Several color invariants were evaluated and performance improvements on the image classification task were demonstrated in the day-night domain adaptation setting by [9].

The zero-shot setting contrasts the more typical approach of unsupervised domain adaptation, where unlabeled samples from the test set are exploited during training to promote the emergence of "deep" features that are invariant with respect to the shift between the domains [4]. The use of the zero-shot setting is motivated by [9] by its removal of any reliance on the availability of test data.

In this paper we assume that test data is readily available, which enables the use of unsupervised domain adaptation methods. We use a well-researched method for using unsupervised domain adaptation with a unified architecture using standard backpropagation training [5]. We show that this method and CIConv similarly aim to reduce a day-night distribution shift and we show that both methods can be implemented in a single architecture. We therefore *compare* the performance of image classification for day-night domain adaptation in the zero-shot and the unsupervised setting, *and* explore the effectiveness of using CIConv and unsupervised domain adaptation together. We formulate the following research question:

What is the effectiveness of CIConv in an unsupervised setting for day-night domain adaptation?
With two subquestions being:

- How does the zero-shot setting with CIConv compare to the unsupervised setting (without CIConv)?
- What is the effect of using CIConv in an unsupervised setting?

We have the following contributions: (i) We evaluate and compare the zero-shot and the unsupervised setting for day-

night domain adaptation; (ii) We evaluate the effectiveness of using CIConv in both settings; and (iii) show and discuss improved performance over both CIConv and unsupervised domain adaptation by introducing their combined setting.
All datasets and code will be made available on our project page.[1]

2 Related work

Unsupervised Domain Adaptation The task of training a model on a source domain in such a way that it performs acceptably on a target domain that is different but related is known as domain adaptation [4]. This is used as an alternative to the process of gathering and annotating sufficient training samples to cover the target domain, which can be non-trivial or even impossible. This paper addresses the domain shift of a training dataset with images only taken in the daytime and a testing dataset containing images in the nighttime. As we assume unlabeled nighttime samples are available during training, we focus on unsupervised domain adaptation (UDA) methods.

Interest in UDA has surged in recent years, resulting in the emergence of many new algorithms. Large-scale experimentation has been done by [12] on many UDA algorithms known as in 2021. Because of the assumption of UDA that there are no labeled samples from the target domain, there is no straightforward way of properly comparing UDA algorithms, but [12] shows the difference in the accuracies to be smaller than previously thought or in some cases even insignificant.

Since we are not particularly interested in the performance of the UDA method itself but rather in whether a UDA method can be combined with color invariants to improve performance, we will pick a well-known method that will be simple to implement in combination with CIConv. [4] describes an approach that involves augmenting any feed-forward model with some standard layers and a simple new gradient reversal layer, resulting in a domain-adversarial neural network (DANN). The DANN performs feature learning, domain adaptation, and classifier learning in a unified architecture with a single learning algorithm (backpropagation), we therefore consider this approach to be suitable for this research.

Color invariants To improve invariance to illumination changes in computer vision, research has been done on physics-based reflection models. Early work includes invariants derived from the Kubelka-Munk reflection model [9,10], from which invariant edge detectors have been derived by [6]. Color invariants have been widely used in classical computer vision applications, but research about their use in deep learning settings is limited.

Therefore, [9] has introduced a learnable color invariant CNN layer (CIConv), evaluated several color invariants in a day-night domain adaptation setting, and demonstrated improved performance on tasks related to autonomous driving. In this paper we will use CIConv with the color invariant [9] reported as performing the best in an unsupervised domain adaptation setting.

[1] https://gitlab.tudelft.nl/attilalengyel/brp-ciconv

3 Method

3.1 CIConv

A learnable Color Invariant Convolution (CIConv) layer as introduced by [9] can be used as the input layer to any CNN to transform the input to a domain invariant representation. As shown by [9], CIConv reduces the distribution shift between the source and target domain, improving the performance on the target domain.

CIConv implements color invariant edge detectors from [6], derived from the image formation model based on the Kubelka-Munk theory [8] for material reflections. The Kubelka-Munk theory for material reflections describes the spectrum of light E reflected from an object in the viewing direction as

$$E(\lambda, x) = e(\lambda, x)((1-\rho_f(x))^2 R_\infty(\lambda, x) + \rho_f(x)) \quad (1)$$

where x denotes the spatial location on the image plane, λ the wavelength of the light, $e(\lambda, x)$ the spectrum of the light source, R_∞ the material reflectivity and ρ_f the Fresnel reflectance coefficient. The partial derivatives of Eq. 1 with respect to x and λ, are denoted by E_x and E_λ.

By making certain assumptions to simplify parts of Eq. 1, various invariant representations can be derived that can be implemented in the CIConv layer. Five invariant representations, each invariant to a different combination of types of illumination changes, were evaluated and compared in the context of day-night domain adaptation by [9]. We will use the color invariant reported by [9] as performing the best, defined as

$$\begin{aligned} W &= \sqrt{W_x^2 + W_{\lambda x}^2 + W_{\lambda\lambda x}^2 + W_y^2 + W_{\lambda y}^2 + W_{\lambda\lambda y}^2}, \\ W_x &= \frac{E_x}{E}, W_{\lambda x} = \frac{E_{\lambda x}}{E}, W_{\lambda\lambda x} = \frac{E_{\lambda\lambda x}}{E} \end{aligned} \quad (2)$$

where λ is the spectral derivative, x is the spatial derivative of Eq. 1. Spatial derivatives for the y direction follow directly from the ones given for the x direction.

The Gaussian color model [6] is used to estimate E, E_λ, $E_{\lambda\lambda}$, as

$$\begin{bmatrix} E(x,y) \\ E_\lambda(x,y) \\ E_{\lambda\lambda}(x,y) \end{bmatrix} = \begin{bmatrix} 0.06 & 0.63 & 0.27 \\ 0.3 & 0.04 & -0.35 \\ 0.34 & -0.6 & 0.17 \end{bmatrix} \begin{bmatrix} R(x,y) \\ G(x,y) \\ B(x,y) \end{bmatrix} \quad (3)$$

where x, y determine the pixel location in the image. The spatial derivatives E_x and E_y are then calculated by convolving E with a Gaussian derivative kernel g with standard deviation σ, i.e.

$$E_x(x, y, \sigma) = \sum_{t \in Z} E(t, y) \frac{\partial g(x-t, \sigma)}{\partial x} \quad (4)$$

and similarly for E_y, $E_{\lambda x}$, $E_{\lambda\lambda x}$, $E_{\lambda y}$, $E_{\lambda\lambda y}$. CIConv is defined as

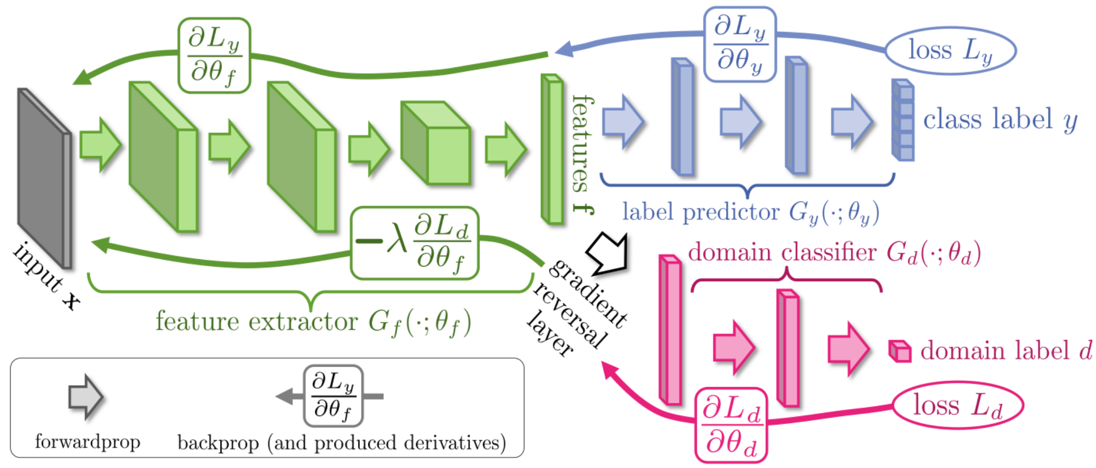

Figure 1: The architecture to perform the *unsupervised domain adaptation by backpropagation* method we use to perform our experiments, as proposed by [4]. It is composed of a standard feed-forward architecture, consisting of a feature extractor (green) and a label predictor (blue), augmented by a *gradient reversal layer (GRL)* that connects a *domain classifier* (red) to the feature extractor (green).

$$CIConv(x,y) = \frac{\log\left(CI^2(x,y,\sigma=2^s)+\epsilon\right)-\mu_{\mathcal{S}}}{\sigma_{\mathcal{S}}} \quad (5)$$

with CI being a color invariant, in our case the color invariant W defined as Eq. 2, $\mu_{\mathcal{S}}$ and $\sigma_{\mathcal{S}}$ the sample mean and standard deviation over $\log(CI^2+\epsilon)$, and ϵ a small term added for numerical stability.

The σ parameter in Eq. 4 determines the scale at which an image is convolved; a small σ will result in a detailed edge map but is more sensitive to noise, a large σ is more robust to noise but is more likely to miss important edges. Experiments by [9] have shown that making the scale parameter $\mathcal{S}$ in Eq. 5 learnable results in σ converging to a task-specific optimal value.
The resulting Color Invariant Convolution (CIConv) is a layer that outputs a single channel representation onto which subsequent convolutional layers can be stacked.

3.2 Unsupervised Domain Adaptation by Backpropagation

We use an approach to unsupervised domain adaptation of deep feed-forward architectures proposed by [4]. This approach achieves adaptation by aligning the distributions of features across the two domains through standard backpropagation training.

It works by having a model predicting for each input x its corresponding label y *and* its domain label $d \in \{0,1\}$. d is a binary variable indicating whether an input is from the labeled day training set ($d=0$) or from the unlabeled night test set ($d=1$). We define three mappings: a feature extractor G_f with parameters θ_f, a label predictor G_y with parameters θ_y and a domain classifier G_d with parameters θ_d. G_f and G_y together form a standard feed-forward architecture; we aim to minimize the label prediction loss on the day training set and therefore their parameters θ_f and θ_y are optimized during training. This leads to discriminativeness of the features which leads to good performance on the samples of the day domain. At the same time, we want to make the features domain-invariant to also ensure good performance on the samples of the night domain. This is done by simultaneously seeking:

1. Parameters θ_f that *maximize* the loss of the domain classifier G_d; maximum loss meaning that the two feature distributions are indistinguishable and therefore predicting the domain label $d \in \{0,1\}$ is as accurate as randomly guessing, provided that the parameters θ_d have been optimally trained.

2. Parameters θ_d that *minimize* the loss of the domain classifier G_d; minimized loss meaning that G_d is trained to discriminate between the two feature distributions in an optimal way.

Altogether this leads to the desired promoted emergence of features that are both discriminative for the image classification task and invariant with respect to the day-night domain shift.

Any feed-forward architecture that is trainable by backpropagation can be augmented with a domain classifier and a special *gradient reversal layer* to implement this idea, as proposed by [4]. This gradient reversal layer connects the feature extractor to the domain classifier and reverses the gradient during backpropagation training by multiplying it by a negative scalar. The resulting architecture is called a *Domain-Adversarial Neural Network (DANN)*.

Since this method leaves the input unchanged during forward propagation and CIConv implemented in the zero-shot setting conforms the type of architecture this method can be used on, we can conclude CIConv can be implemented in combination with this method of unsupervised domain adaptation.

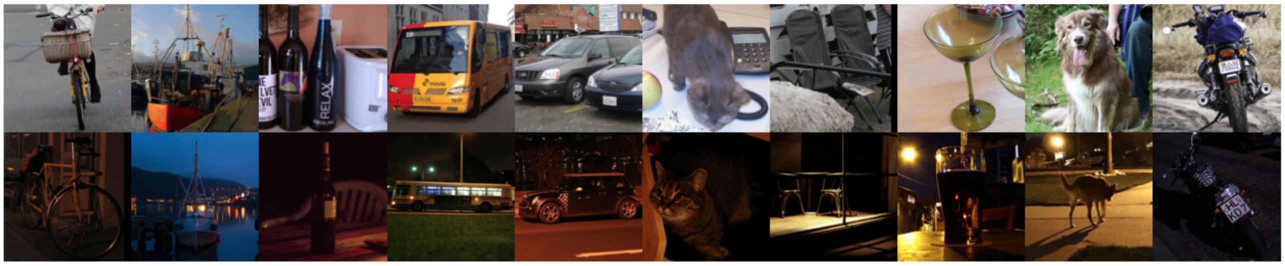

Figure 2: Samples from day (source domain) test set (top) and the night (target domain) test set (bottom) of the CODaN dataset [9] we used in our experiments.

4 Experiments

We perform experiments in both the zero-shot setting and the unsupervised domain adaptation setting. For the zero-shot setting, we run the same classification experiments as performed by [9] but with slightly adjusted settings for consistency across all of our experiments. These zero-shot experiments are done to gather baseline results and to further verify the performance improvements for classification by CIConv reported by [9].

4.1 Zero-shot setting

We use the Common Objects Day and Night (CODaN) dataset, as presented by [9]. It consists of natural images from 10 common object classes recorded in both day and night time. It contains a daytime training set of 1,000 samples per class, a daytime validation set of 50 samples per class, and separate day and night test sets of 250 samples per class. We disregard 175 samples per class of the night test set that will later be used for experiments in the unsupervised domain adaptation setting to allow for fair comparisons of the test scores. CODaN is composed of parts of the ImageNet [3], COCO [10], and ExDark [11] datasets. Day and night test set samples are shown in Fig 2.

A baseline ResNet-18 and its color invariant version with the CIConv layer with invariant W will be used for training. We use the implementation of the classification experiments by [9]. The training is done for 175 epochs with a batch size of 64, using SGD with momentum 0.9, weight decay 1e-4, and an initial learning rate of 0.05 with stepwise reduction by factor 0.1, step size 50. Data augmentation is performed in the form of random horizontal flips and random rotations. We also apply random brightness, contrast, hue, and saturation augmentations. For the sake of memory usage, we reduce the resolution of the entire dataset by half, from 224x224 to 112x112. This will affect the results as opposed to using the original resolution but will still allow for fair comparisons to be made. Table 1 shows the classification accuracies of the two models averaged over three runs. To show how the resizing affects the performance, the classification accuracies of both models on the CODaN dataset with its original resolutions are also shown.

Method	**Day**	**Night**
Without CIConv (resized)	68.9 ± 0.3	38.3 ± 0.4
With CIConv (resized)	69.8 ± 0.7	49.4 ± 0.3
Without CIConv (no resizing)	80.7 ± 0.9	49.3 ± 1.1
With CIConv (no resizing)	81.8 ± 0.3	61.6 ± 0.6

Table 1: CODaN classication accuracies of a ResNet-18 architecture averaged over three runs.

4.2 Unsupervised Domain Adaptation

We again use the CODaN dataset. For the unlabeled target distribution needed to perform unsupervised domain adaptation during training, we use the 175 samples of each class of the night test set that we disregarded in 4.1.

A Domain-Adversarial Neural Network (DANN) consisting of the baseline ResNet-18 architecture used in 4.1, extended with a domain classifier connected to the feature extractor via a gradient reversal layer will be used for training. Since any feed-forward architecture that is trainable by backpropagation can be extended to perform unsupervised domain adaptation by backpropagation [4], the CIConv layer with invariant W can be implemented into the DANN in the same way as the color invariant version of the ResNet-18 in 4.1.

We use the DANN implementation of the unsupervised domain adaptation by backpropagation method [4] by [7] with adjustments to work with the CODaN dataset and to allow for the implementation of CIConv.

Since the models use significantly more memory for training compared to the models from experiment 4.1, our environment does not allow training on CODaN in full resolution, hence the resizing by half in both experiment 4.1 and this experiment. Furthermore, the same settings are used as in 4.1. The same data augmentation is also performed on the night test set that is used for training. Table 2 shows the classification accuracies of the two models, averaged over three runs.

Method	Day	Night
Without CIConv (resized)	68.4 ± 1.2	49.2 ± 1.5
With CIConv (resized)	69.7 ± 0.5	58.2 ± 0.4

Table 2: CODaN classication accuracies of a DANN ResNet-18 architecture averaged over three runs.

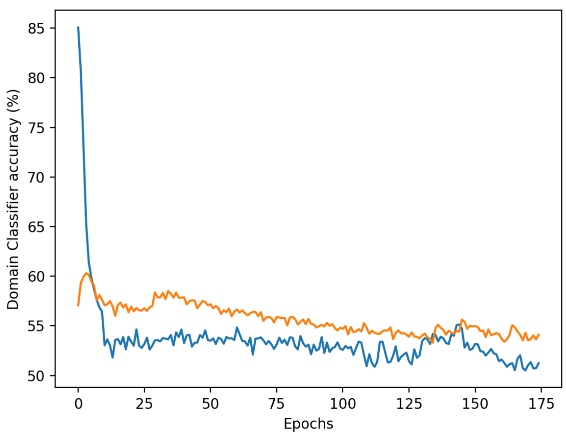

Figure 3: Accuracy of the *domain classifier* during the training of a ResNet-18 DANN on the CODaN dataset with CIConv implemented (orange) and without CIConv implemented (blue).

5 Discussion

We performed experiments to compare the performance of image classification for day-night domain adaptation in the zero-shot and the unsupervised setting, and explore the effectiveness of using CIConv and unsupervised domain adaptation together. The effectiveness of CIConv in the zero-shot setting was already demonstrated by [9] and this was confirmed by our experiments (see Table 1). We showed that the unsupervised domain adaptation (UDA) method we used, chosen based on representativeness of the UDA research field and consistency across our experiments, performed similar to CIConv in the zero-shot setting (see Table 1 and 2). Interesting to see was that our experiments of CIConv implemented in the unsupervised domain adaptation setting performed significantly better over the other experiments (see Table 2).

The foundation of CIConv is based on certain assumptions of lighting conditions that most natural scenes do not meet; e.g. reflections being purely matte, all materials being non-transparent and the scene having a single, spatially uniform light source. This is done for simplification of Eq. 1 to derive several color invariant edge detectors that can be implemented in CIConv. These color invariant edge detectors, each having different invariance properties, were evaluated in the zero-shot setting for day-night domain adaptation by [9], the best-performing was chosen to be used in our experiments. Fig. 3 shows that the domain classifier is still able to detect a small domain shift when CIConv is used, at the beginning of training with an accuracy of around 60% which gradually drops towards 55%. This shows how CIConv is not able to transform the input to a fully domain invariant representation, since that would lead to the domain classifier performing equally well as a random binary number generator (see 3.2). It is unclear to what extent the gradual improvement is caused by CIConv learning the optimal scale parameter or UDA being able reduce the remaining domain discrepancy. However, the significantly better results over CIConv in the zero-shot setting imply that UDA plays a significant role in minimizing the discrepancy between feature distributions for the day and night domain. Further experimentation, for example running the experiments with different invariants or parameters for CIConv, could be done to further evaluate the role of UDA when CIConv is used.

Fig. 3 shows a surprisingly fast drop of the accuracy of the domain classifier in the experiments for the unsupervised setting without CIConv, from high accuracies for the first 5-10 epochs to ranging between 50-55% for the rest of the training. This would mean that UDA is able to quickly minimize the discrepancy between feature distributions for the day and night domain and proceeds training in a standard way by minimizing the label prediction loss (see 3.2). However, the results still show a significantly lower performance on the night domain. This leads us to question the representativeness of the test samples we used during training for the night domain as a whole. Due to time constrains we used a relatively small dataset, resulting in a small amount of samples from the target domain that could be used for training the models in the unsupervised domain adaptation setting from 4.2. This could mean that the samples from the night domain for training were lacking representativeness for the samples from the night domain for testing; that a significant *shift* was present between train and test data from the target domain. This leads us to be rather inconclusive about the effectiveness of CIConv in the unsupervised domain adaptation setting; it could be be that UDA will benefit from using a larger dataset to an extent that it will outperform the combination of UDA and CIConv. Further experimentation with larger datasets is needed to draw more conclusive remarks.

Due to memory limitations of our experimentation environment we could not run the UDA experiments on CODaN without resizing the images. We ran experiments in the zero-shot setting for both the resized and original CODaN dataset to show that the resizing leads to significantly lower results (see Table 1). However, CIConv showing a similar ratio between the performances on the day and night domain in both experiments leads us to assume that resizing has a negligible influence on the effectiveness of CIConv. Further experimentation without resizing is needed to prove this assumption.

Unsupervised domain adaptation in combination with CIConv is a promising method for day-night domain adaptation. Although we remain inconclusive about to what extent CIConv is effective in the unsupervised setting, we showed promising performance improvements with our experiments. We therefore hope that this paper inspires future research on combining physics priors with unsupervised domain adaptation methods.

6 Responsible Research

The research field of deep learning is prone to ethical problems and hence requires proper discussion, as will be done in this section. This paper discusses methods for improving the potential of neural networks to be used in safety-critical applications, such as autonomous driving. It is important to note that the discussed methods are not ready to be directly implemented and used in any safety-critical application without proper tweaking and testing. This paper only demonstrates (improvements in) performance and does not give any guarantee of safety when being used in the real world. Many works from other authors are used to conduct this research, for which proper credit is important. The data used in this research consists of a single dataset, the Common Objects Day and Night (CODaN) dataset introduced by [9] which is publicly available and composed of other public and well-referenced datasets. All of the code used to run experiments from other repositories has been properly credited and referenced.

6.1 Reproducibility

The dataset used in this research, CODaN, is publicly available for anyone to use and its repository contains instructions for implementation. All of the code used to run experiments is available on our repository1. To reproduce the experiments with the same or similar results we refer to 4.1 and 4.2, in which relevant parameters, data augmentation settings, etc. are discussed. Since the majority of the code is based on other repositories, we refer to these repositories if anything does not work as intended.

References

[1] Mahmoud Afifi and Michael S Brown. What else can fool deep learning? addressing color constancy errors on deep neural network performance. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 243–252, 2019.

[2] Dengxin Dai and Luc Van Gool. Dark model adaptation: Semantic image segmentation from daytime to nighttime. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pages 3819–3824. IEEE, 2018.

[3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[4] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *International conference on machine learning*, pages 1180–1189. PMLR, 2015.

[5] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1):2096–2030, 2016.

[6] J-M Geusebroek, Rein Van den Boomgaard, Arnold W. M. Smeulders, and Hugo Geerts. Color invariance. *IEEE Transactions on Pattern analysis and machine intelligence*, 23(12):1338–1350, 2001.

[7] Bo Fu Mingsheng Long Junguang Jiang, Baixu Chen. Transfer-learning-library. https://github.com/thuml/Transfer-Learning-Library, 2020.

[8] Paul Kubelka. Ein beitrag zur optik der farbanstriche (contribution to the optic of paint). *Zeitschrift fur technische Physik*, 12:593–601, 1931.

[9] Attila Lengyel, Sourav Garg, Michael Milford, and Jan C van Gemert. Zero-shot day-night domain adaptation with a physics prior. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4399–4409, 2021.

[10] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[11] Yuen Peng Loh and Chee Seng Chan. Getting to know low-light images with the exclusively dark dataset. *Computer Vision and Image Understanding*, 178:30–42, 2019.

[12] Kevin Musgrave, Serge Belongie, and Ser-Nam Lim. Unsupervised domain adaptation: A reality check. *arXiv preprint arXiv:2111.15672*, 2021.

[13] Eduardo Romera, Luis M Bergasa, Kailun Yang, Jose M Alvarez, and Rafael Barea. Bridging the day and night domain gap for semantic segmentation. In *2019 IEEE Intelligent Vehicles Symposium (IV)*, pages 1312–1318. IEEE, 2019.

[14] Mei Wang and Weihong Deng. Deep visual domain adaptation: A survey. *Neurocomputing*, 312:135–153, 2018.

[15] Markus Wulfmeier, Alex Bewley, and Ingmar Posner. Addressing appearance change in outdoor robotics with adversarial domain adaptation. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1551–1558. IEEE, 2017.