International Journal of Computer Vision (2023) 131:752–771
https://doi.org/10.1007/s11263-022-01718-1

S.I. : BMVC'20

DESC: Domain Adaptation for Depth Estimation via Semantic Consistency

Adrian Lopez-Rodriguez[1] · Krystian Mikolajczyk[1]

Received: 27 August 2021 / Accepted: 7 November 2022 / Published online: 15 December 2022

Abstract
Accurate real depth annotations are difficult to acquire, needing the use of special devices such as a LiDAR sensor. Self-supervised methods try to overcome this problem by processing video or stereo sequences, which may not always be available. Instead, in this paper, we propose a domain adaptation approach to train a monocular depth estimation model using a fully-annotated source dataset and a non-annotated target dataset. We bridge the domain gap by leveraging semantic predictions and low-level edge features to provide guidance for the target domain. We enforce consistency between the main model and a second model trained with semantic segmentation and edge maps, and introduce priors in the form of instance heights. Our approach is evaluated on standard domain adaptation benchmarks for monocular depth estimation and show consistent improvement upon the state-of-the-art. Code available at https://github.com/alopezgit/DESC.

Keywords Depth estimation · Domain adaptation · Semantic consistency · Image translation

1 Introduction

State-of-the-art depth estimation methods are capable of inferring an accurate depth map from a monocular image by relying on deep learning methods, which require a large amount of data with annotations (Fu et al., 2018; Laina et al., 2016). Annotations in the form of precise depth measurements are typically provided by special tools such as a LiDAR sensor (Geiger et al., 2012) or structured light devices (Silberman et al., 2012). Thus, obtaining depth annotations is costly and time-consuming. Much research has focused on developing methods not relying on directly acquired depth annotations by leveraging stereo (Godard et al., 2017; Garg et al., 2016) or video sequences (Godard et al., 2019; Casser et al., 2019; Yin & Shi, 2018) for self-supervision. These research directions have shown promise, but a stereo pair or video sequence may not always be available in existing datasets. The use of synthetic data provides a way to obtain a large amount of accurate ground truth depth in a fast manner. However, synthetic data and real data have usually a domain gap, mainly due to the difficulty of generating photorealistic synthetic images. In that direction, domain adaptation techniques (Nath Kundu et al., 2018; Zheng et al., 2018) can help to transfer the models trained on an annotated source dataset $\mathcal{S}$ to a target dataset $\mathcal{T}$, reducing the burden of training a model for a new environment or camera.

Research results have shown that the domain gap for semantic segmentation and instance detection can be reduced by introducing depth information during training (Liu et al., 2019; Vu et al., 2019; Chen et al., 2019c; Saha et al., 2021). A different direction, which leverages semantic information to reduce the domain gap in depth estimation, has been less studied and mainly in multi-task scenarios (Atapour-Abarghouei & Breckon, 2019; Kundu et al., 2019). The high-level structure of the scene, which is given in a semantic map, is a compact representation with lower domain gap compared to RGB images (e.g., textures or illumination in RGB images are highly domain dependant) (Zhou et al., 2020) and gives information about the geometry of the scene. Humans, for example, use several semantic cues to estimate depth, e.g., smoothness of depth values in an object instance (Chen et al., 2019b), relative size of known objects in the image or vertical position of instances in the image (Dijk & Croon,

Communicated by Martin Fergie.

This research was supported by UK EPSRC project EP/S032398/1.

✉ Adrian Lopez-Rodriguez
al4415@imperial.ac.uk

Krystian Mikolajczyk
k.mikolajczyk@imperial.ac.uk

[1] Imperial College London, London, UK

2019). In addition, existing datasets with semantic annotations are large and diverse in scenes as well as cameras used, hence models trained on these diverse semantic datasets are capable of generalizing to different settings (Lambert et al., 2020). Several works (Li & Snavely, 2018; Casser et al., 2019) have shown that using pretrained models to obtain semantic annotations can also bring improvements in the depth estimation task. Motivated by these findings, we exploit readily-available panoptic segmentation models as guidance to bridge the gap between two different domains and to improve monocular depth estimation.

Domain adaptation approaches benefit from pseudo-labelling (Chen et al., 2019a; Saito et al., 2017) and consistency of predictions in the source and target domains (Zhao et al., 2019; Chen et al., 2019d). Therefore, we propose an approach that leverages semantic annotations to enforce consistency for depth estimation between the two domains, and to provide depth pseudo-labels to the target domain by using the size of the detected objects. Figure 1 shows an overview of the task. Our main contributions are: (1) the proposal of an approach to form depth pseudo-labels in the target domain by using object size priors, which are learnt in an instance-based manner in the annotated source domain; (2) the introduction of a consistency constraint with predictions from a second model trained on high-level semantics and low-level edge maps; (3) state-of-the-art results in the task of monocular depth estimation without self-supervision with domain adaptation from VirtualKITTI (Gaidon et al., 2016) to KITTI (Geiger et al., 2012).

In this paper, we extend the original Depth Estimation via Semantic Consistency (DESC) (Lopez-Rodriguez & Mikolajczyk, 2020) work by expanding the experimental section to add new settings and ablation studies, and also by including recent domain adaptation works. The new experiments contain improvements over the original DESC by combining our semantic consistency modules with advancements in image transfer modules, and also by using an ImageNet-pretrained encoder following state-of-the-art depth estimation works. Furthermore, we extend the evaluation of DESC by adding an analysis of the generalization capabilities on Make3D, results in a semi-supervised setting proposed by past works, and a largely extended assessment of its performance both quantitatively and qualitatively.

2 Related Work

2.1 Monocular Depth Estimation

Self-Supervision Early depth estimation methods relied on supervised training, using annotations from LiDAR (Geiger et al., 2012) or structured light scanners (Silberman et al., 2012). Due to the difficulty of obtaining depth annotations, several works have focused on using either stereo pairs or video self-supervision. Xie et al. (2016) regressed a discretized disparity map and used a pixel-wise consistency loss with a second camera view, and Garg et al. (2016) extended it to predict continuous depth values. The accuracy was further improved in Monodepth (Godard et al., 2017) by forcing the network to predict from a single image both left and right disparities and adding a consistency term. A stereo pair was used in Luo et al. (2018) to supervise a model that synthesized the right view from the left image, and then processing both views by a stereo-matching network. Other notable approaches include the use of adversarial techniques and cycle-consistency (Pilzer et al., 2019a, b). Stereo images are not always available, hence video self-supervision has also been researched. Simultaneous learning of depth and pose was addressed in Zhou et al. (2017), which given three video frames, projected the $t+1$ and $t-1$ views to the reference view t. Joint pose, depth and optical flow learning was proposed in GeoNet (Yin & Shi, 2018), and Monodepth2 (Godard et al., 2019) focused on improving the pixel reprojection loss and the multiscale loss.

2.2 Depth and Semantic Information

Depth and semantic information have been utilized simultaneously to improve depth estimation. The two predominant trends involve either using a multi-task approach to improve the depth predictor features, or using the semantic masks to regularize (e.g., smoothness, edge alignment) and/or filter (e.g., dynamic objects for self-supervision) the depth maps.

Multi-task learning of depth and semantic tasks has been proposed in multiple works. Mousavian et al. (2016) trained a single network for both semantic and depth prediction in a multi-task manner by using a shared backbone and task-specific layers. In that direction, Chen et al. (2019b) trained a network capable of selecting between depth or semantic segmentation output by only changing an intermediate task layer. Several works (Jiao et al., 2018; Choi et al., 2020; Li et al., 2020; Zhang et al., 2018) proposed novel units to share information between the two tasks, which improved the depth features. Atapour-Abarghouei and Breckon (2019) assumed the availability of temporal information in training and test time to fuse multiple frames for depth and semantic segmentation prediction. Guizilini et al. (2020) used a pretrained semantic segmentation network to guide the feature maps of the depth network using pixel-adaptive convolutions.

Regularization of Depth with semantic information has also been done to improve depth prediction quality. In that direction, MegaDepth (Li & Snavely, 2018), a diverse Structure-from-Motion and Multi-View Stereo depth dataset collected from the internet, used semantic information to filter spurious depth values and to define ordinal labels.

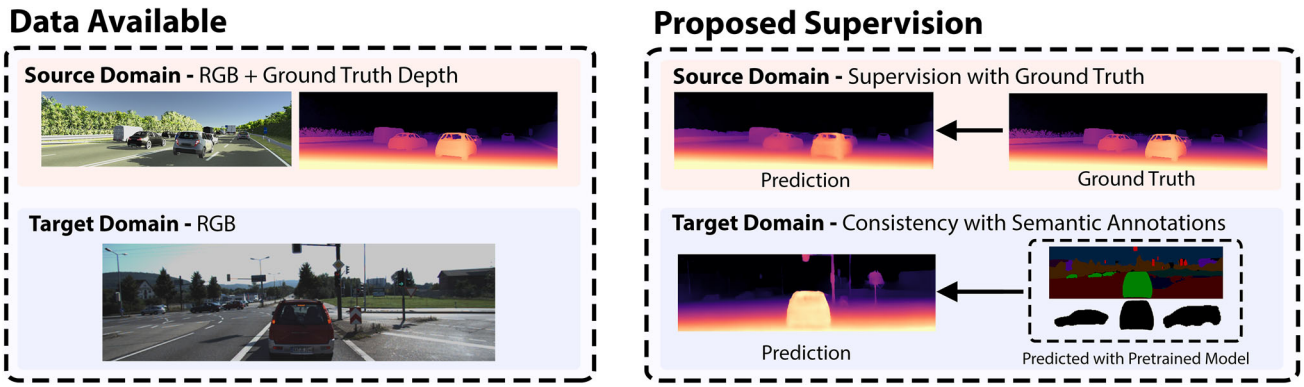

Fig. 1 Overview of the data available and proposed supervision. The source domain $\mathcal{S}$ contains both RGB and ground truth depth data, and the target domain $\mathcal{T}$ contains RGB data only. We train a depth estimation model to achieve high performance in $\mathcal{T}$ by leveraging semantic annotations to introduce semantic consistency in $\mathcal{T}$. The semantic annotations are obtained using a panoptic segmentation model trained with external data

Some works used precomputed object instances masks by filtering the dynamic objects from the photometric loss in a video self-supervision setting (Casser et al., 2019; Meng et al., 2019; Kirillov et al., 2020). Related to our work, Struct2Depth (Casser et al., 2019), apart from filtering dynamic objects from the photometric loss, also used predicted instance masks to impose a object size-depth constraint by learning a single object height for all the *car* instances. Kirillov et al. (2020) trained a semantic segmentation branch used to detect dynamic objects, filtering out those that are moving while learning from those that are static (e.g., parked cars). Zhu et al. (2020) used semantic maps to regularize the depth edge of object instances using a morphing operator and a consistency loss, which aimed to tackle the bleeding artifacts in a stereo supervised method. Following this depth edge regularization approach, Saeedan and Roth (2021) employed panoptic maps to force a depth discontinuity in the instance edges, and also for stereo consistency, which alleviates some issues with photometric consistency (e.g., non-Lambertian surfaces).

2.3 Domain Adaptation

Domain adaptation is attracting an increasing attention due to the lack of a sufficient volume of annotated data for supervised training. It showed some success in areas such as classification (Saito et al., 2017; Tzeng et al., 2017) and semantic segmentation (Chen et al., 2019d; Tsai et al., 2018). Popular approaches include style adaptation of the source data to match the target data (Hoffman et al., 2018; Lopez-Rodriguez et al., 2020), adversarial approaches to match either the features (Ganin & Lempitsky, 2015; Tzeng et al., 2017) or outputs (Tsai et al., 2018) of the domains, and using pseudo-labels (Chen et al., 2019a; Saito et al., 2017).

Depth Estimation Image translation techniques have been widely used for domain adaptation in depth estimation tasks due to its success in decreasing the domain gap (Atapour-Abarghouei et al., 2018; Zhao et al., 2019; Zheng et al., 2018; PNVR et al., 2020; Cheng et al., 2020). Atapour-Abarghouei et al. (2018) generated synthetic data using the video-game GTA V and used a cycle-consistency image-transfer approach, which also added computational burden by translating the target images during inference. Our base DESC approach (Lopez-Rodriguez & Mikolajczyk, 2020) uses the strategy presented in T^2Net (Zheng et al., 2018), which performs image translation without a cycle-consistency loss, reducing the complexity and the number of networks needed, and additionally removing any need for inference-time translation contrary to Atapour-Abarghouei et al. (2018). Several works have built upon T^2Net (PNVR et al., 2020; Cheng et al., 2020). GASDA (Zhao et al., 2019) focused on the scenario where stereo supervision is available in the target domain, and added stereo photometric guidance and depth prediction consistency between original and style-transferred target domain images. GASDA (Zhao et al., 2019) also increased the test-time complexity by averaging the predicted depth map of the original and the style-transferred target images. SharinGAN (PNVR et al., 2020) modified T^2Net by transforming both the target and source images into a shared intermediate domain using a shared generator, which improved results also at the cost of increased complexity at test time. Another improvement over T^2Net was given by S^3Net (Cheng et al., 2020), which focused on the combination of synthetic ground truth, predicted semantic maps (also used at test time) and real video self-supervision. In S^3Net (Cheng et al., 2020), extra constraints were imposed in the translation step, namely: multi-frame photometric consistency and semantic consistency using segmentation maps. Contrary to these methods, AdaDepth (Nath Kundu et al.,

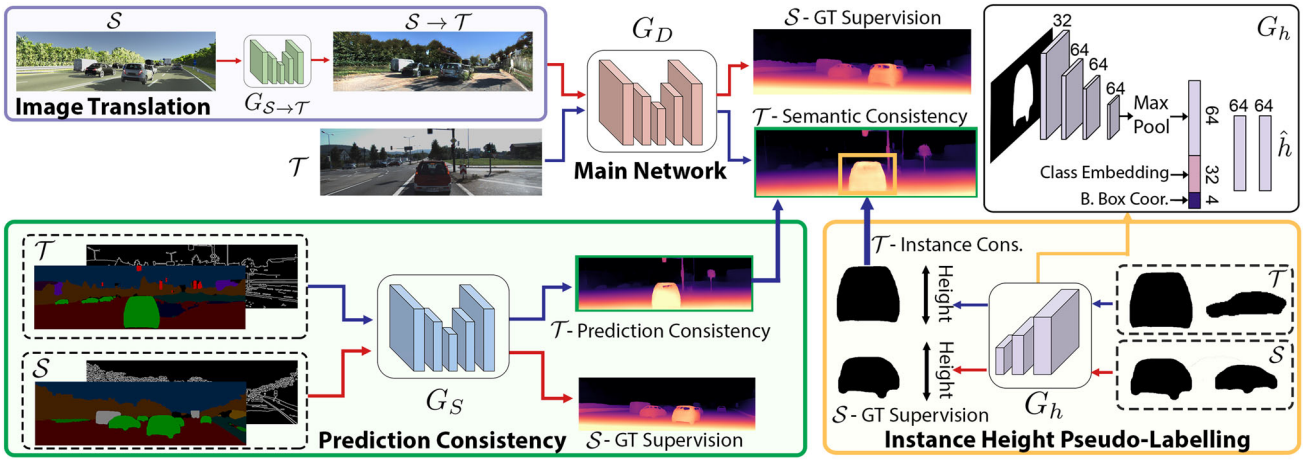

Fig. 2 Overview of the approach. We train a depth estimation network G_D with both target $\mathcal{T}$ and source $\mathcal{S}$ images. Source images are adapted to the style of the target images. For $\mathcal{S}$, we use ground truth supervision, while we enforce consistency with semantic information in $\mathcal{T}$. The consistency is enforced with (1) predictions from a second network G_S trained with edges and semantic maps as input, and (2) depth pseudo-labels formed using an instance height $\hat{h}$ predicted by G_h. Both G_S and G_h are trained using ground truth data from $\mathcal{S}$. The architecture of G_h is given in the top right. We use ReLU between the layers of G_h

2018) did not use any image translation and employed instead an adversarial approach to align both output and feature distributions between the source and target domain, along with a feature consistency module to avoid mode collapse. MonoDEVSNet (Gurram et al., 2021), which similarly to S^{3}Net (Cheng et al., 2020) used video self-supervision for the real domain, focused on absolute scale depth prediction and employed semantic maps for source data weighting. MonoDEVSNet also used a feature adaptation approach similarly to AdaDepth, but using instead a gradient-reversal layer. In a multi-task setup, Kundu et al. (2019) proposed a cross-task distillation module and contour-based content regularization to extract feature representations with greater transferability.

Beyond Unsupervised Domain Adaptation In a semi-supervised domain adaptation task, ARC (Zhao et al., 2020) also used image translation, in this case to remove the clutter from the real domain before depth prediction. ARC follows the hypothesis that, compared to the cleaner synthetic images, the clutter and novel objects in real data is the reason for the domain gap. In a domain generalization context, S2R-DepthNet (Chen et al., 2021) used only synthetic data to train a model capable of generalizing to unseen real data. S2R-DepthNet uses two extra networks to transform both the synthetic and real data into images containing mostly structural edges needed for depth estimation, thus removing unnecessary information (e.g., textures) and reducing the domain gap.

Synthetic Data Several synthetics datasets that can be used for depth estimation have been developed, especially within driving scenarios. Virtual KITTI (Gaidon et al., 2016) provides a synthetic version of KITTI, which was improved in the follow-up Virtual KITTI 2 (Cabon et al., 2020). SYNTHIA (Ros et al., 2016) provides multi-camera images and depth annotations, whereas CARLA (Dosovitskiy et al., 2017) offers a simulated environment where virtual cameras can be placed arbitrarily. In non-driving settings, some synthetic datasets that provide depth annotations are also available (Mayer et al., 2016; Li et al., 2018a).

Positioning of DESC Similarly to T^{2}Net and AdaDepth, and contrary to methods focusing on real domain stereo (GASDA, SharinGAN) or video (MonoDEVSNet, S^{3}Net) self-supervision, we focus on the setting where no self-supervision is available in the target domain, where we report state-of-the-art results. Semantic information is also employed in works concurrent to or newer than the original work in DESC (Lopez-Rodriguez & Mikolajczyk, 2020), specifically for image translation improvement and input augmentation (S^{3}Net) or source data weighting (MonoDEVSNet). Unlike those works, we use semantic information to bring guidance in the output map inspired by consistency-based domain adaptation works (Roy et al., 2019; French et al., 2018; Sajjadi et al., 2016). First, we leverage ideas from Struct2Depth (Casser et al., 2019) to form target domain pseudo-labels, albeit we predict an individual height per instance by using source domain ground truth as guidance. Secondly, motivated by the low domain-gap of segmentation maps, also noted in a concurrent work (Zhao et al., 2020), we force consistency in the output map between two networks with different input representations (RGB, and Semantic+Edges). As we focus on output map guidance, we do not add any extra computationally burden contrary to some of the past methods (e.g., GASDA, SharinGAN or

Fig. 3 Intra-class variation of detected instances, where we show examples variations of the pose of a detected person (top row), missing bike handlebar (middle row) and occlusion effects on a detected car (bottom row)

S2R-DepthNet), and we do not use any semantic information at test time contrary to S^{3}Net. Furthermore, DESC can be combined with different image-transfer variants, as shown in Sect. 4.2 in the improvement achieved using strategies from T^{2}Net, SharinGAN or S2R-DepthNet.

3 Method

In this section, we introduce our domain adaptation for Depth Estimation via Semantic Consistency (DESC) approach. An overview is presented in Fig. 2. During inference we only apply our depth estimation network G_D to our target images, except when using other image translation strategies in Sect. 4.2. Semantic annotations are predicted for our source and target datasets using a panoptic segmentation model (Kirillov et al., 2019b) trained with external data, providing per image detected instances and a semantic segmentation map.

3.1 Pseudo-labelling Using Instance Height

The height of the detected object instances can provide a strong cue for distance estimation, hence we aim to use the detected instances to provide a guidance in the target domain by generating pseudo-labels from the predicted height. To do so, we leverage the work in Struct2Depth (Casser et al., 2019), which used the instance height to deal with moving objects in video self-supervision. Thus, Struct2Depth retrieved an approximate distance to the objects by solving

$$\hat{D} \approx \frac{f \cdot h}{H} \tag{1}$$

where $\hat{D}$ is an approximate distance to the object, f is the focal length in pixels, H is the predicted instance size in pixels and h is the physical height of the object. It is assumed that the entire object instance is placed at a distance $\hat{D}$, that f is known, and that the real object size h is unknown. In Struct2Depth (Casser et al., 2019), the object size was set as a shared learnable parameter $\hat{h}$ for the class *car*, i.e., all of the detected instances of class *car* were assumed to have the same height. We argue that predicting a $\hat{h}$ per object instance rather than class can provide a better height estimate, as it can take into account both intra-class variations and occlusions in the detected instances. Figure 3 shows some examples of cases of intra-class variations in the detected instances, which affect the height H of the detections in pixels, e.g., the obtained H for the bottom-right car only takes into account part of the car due to occlusion effects. A unique predicted height per class cannot correct for those variations, and thus we need to estimate an instance-based physical height $\hat{h}$ to obtain a more accurate depth when using Eq. (1). Furthermore, instead of learning $\hat{h}$ in an unsupervised manner as in Struct2Depth (Casser et al., 2019), we can improve the estimation using source domain data. Therefore, we use a network G_h, with a simple architecture presented in Fig. 2, to predict a $\hat{h}_i$ for an instance i from the dimensions of its bounding box, the detected binary instance mask and the predicted class label. G_h can use the predicted class to learn a range of suitable values of $\hat{h}$ for the detected instance, and then correct for pose variations, occlusions or other intra-class effects by using the bounding box and binary instance mask. We train G_h using labels in the source data by retrieving $h_{GT,i}$, which is the ground truth physical object size for instance i. To retrieve $h_{GT,i}$ we use $h_{GT,i} = \frac{H_i \cdot \hat{D}_{\mathcal{S},i}}{f_{\mathcal{S}}}$, where the instance depth $\hat{D}_{\mathcal{S},i}$ is obtained directly from the depth ground truth. To obtain $\hat{D}_{\mathcal{S},i}$ we use $\hat{D}_i = median(M_{\mathcal{S},i} \odot y_{\mathcal{S}})$, where $M_{\mathcal{S},i}$ is the binary segmentation instance mask for a source domain detected instance i, $\odot$ refers to the Hadamard product, $y_{\mathcal{S}}$ is the ground truth depth, and the median operation is performed only for non-zero values. Thus, G_h is trained on the source domain with $\mathcal{L}_{I,\mathcal{S}} = \frac{1}{n_I} \sum_i |\hat{h}_{\mathcal{S},i} - h_{GT,i}|$, where n_I is the number of detected instances. In the target domain, G_h is used to predict a height $\hat{h}_{\mathcal{T},i}$ for a detected instance i, and then $\hat{h}_{\mathcal{T},i}$ is used to retrieve a depth pseudo-label $\hat{D}_{\mathcal{T},i}$ computed using Eq. (1). We use the depth pseudo-labels $\hat{D}_{\mathcal{T},i}$ to provide supervision for G_D in the target domain using a sum of pixel-wise L_1 losses over all detected instances i,

$$\mathcal{L}_{I,\mathcal{T}} = \frac{\phi}{p_I} \sum_i \left\| \left(\frac{\hat{D}_{\mathcal{T},i}}{\phi} - G_D(x_{\mathcal{T}}) \right) \odot M_{\mathcal{T},i} \right\|_1 \tag{2}$$

where p_I is the sum of non-zero pixels for all the binary segmentation masks $M_{\mathcal{T},i}$, $x_{\mathcal{T}}$ is an image from $\mathcal{T}$ and ϕ is a learnable scalar. The scalar ϕ is used to correct any scale mismatch in the predictions of $G_D(x_{\mathcal{T}})$ due to camera differences between $\mathcal{S}$ and $\mathcal{T}$ (He et al., 2018). When computing $\hat{D}_{\mathcal{T},i}$ we use the focal length $f_{\mathcal{T}}$ of the target domain camera, although as we will show in Sect. 4, ϕ automatically scales the values to the correct range even for unknown $f_{\mathcal{T}}$.

As we use a panoptic segmentation model trained with external data to extract semantic annotations, some of the classes detected may be present in $\mathcal{T}$ but not in $\mathcal{S}$, e.g., *person* in Virtual KITTI→KITTI. For those classes, G_h can also learn an instance-based height prior in an unsupervised manner via consistency with G_D in $\mathcal{L}_{I,\mathcal{T}}$.

3.2 Consistency of Predictions Using Semantic Information

Many works (Roy et al., 2019; French et al., 2018; Sajjadi et al., 2016) have shown that constraining the learning process by requiring consistency in a domain adaptation setting reduces the performance gap. Similar observations have been made in self-supervised learning (Chen et al., 2020), where a contrastive loss is used between different views of the same scene obtained via data augmentation. Following these findings, we enforce consistency between the predictions generated by our main depth estimation network, G_D, and a secondary network, G_S. Instead of using data augmentation techniques to generate another view to input to G_S, we aim to use low domain-gap modalities to increase the generalization ability of the network. Hence, input data x_{Sem} is formed by two channels that have a low domain gap: a semantic segmentation map and an edge map.

Semantic Structure A semantic segmentation map provides information on the high-level structure of the scene, and this high-level structure helps to predict the depth structure. Similarly to the observation made by concurrent work (Zhou et al., 2020) to our original DESC work (Lopez-Rodriguez & Mikolajczyk, 2020), we notice that datasets are more similar in their high-level structures or presented semantic scenes compared to their RGB similarity due to differences on the quality of textures, illumination or models, among others. The semantic segmentation map is introduced in the form of an integer corresponding to the semantic class label, as we experimentally found it to yield better performance than one-hot encoding.

Edge Map Deep learning networks tend to use texture cues (Geirhos et al., 2019) for predictions. We use an edge map to reduce the impact of the texture differences between domains, and to provide a different data modality to the network. Furthermore, edge maps include information about the shapes of objects that is valuable in depth related tasks (Hu et al., 2019; Huang et al., 2019). Edges also give complementary information to the semantic maps and, compared to RGB images, present less variation and need less adaptation in domains with semantically similar scenes.

Consistency As both networks G_D and G_S receive different input modalities, forcing consistency between them for the predictions of the target domain can significantly increase the target-domain performance of both models. We propose to supervise G_S with source domain depth ground truth $y_{\mathcal{S}}$ by using a pixel-wise L_1 loss, $\mathcal{L}_{Con,\mathcal{S}}$, and then force consistency of predictions in the target domain via $\mathcal{L}_{Con,\mathcal{T}}$. Then, assuming N is the total number of pixels,

$$\begin{aligned}\mathcal{L}_{Con,\mathcal{S}} &= \frac{1}{N}\|G_S(x_{Sem,\mathcal{S}}) - y_{\mathcal{S}}\|_1 \\ \mathcal{L}_{Con,\mathcal{T}} &= \frac{1}{N}\|G_D(x_{\mathcal{T}}) - G_S(x_{Sem,\mathcal{T}})\|_1\end{aligned} \tag{3}$$

3.3 Training Loss

We now present the modules used in DESC in addition to our semantic consistency losses.

Depth Estimation Loss Our model G_D outputs a multiscale prediction that is supervised using source domain ground truth with $\mathcal{L}_D$, which is a pixel-wise L_1 loss (Zheng et al., 2018; Zhao et al., 2019). The ground truth is resized to match the resolution of each of the maps output by G_D, and specifically the model we use for G_D outputs maps at 4 different scales, where each subsequent map doubles both its width and height. Thus, $\mathcal{L}_D$ is defined as:

$$\mathcal{L}_D = \frac{1}{N}\sum_s \|G_D(x_{\mathcal{S}})_s - y_{\mathcal{S},s}\|_1 \tag{4}$$

where s refers to the scale of the prediction and $y_{\mathcal{S},s}$ is the resized source ground-truth to match the resolution of the prediction $G_D(x_{\mathcal{S}})_s$.

Image Translation has been demonstrated to effectively reduce the domain gap (Zhao et al., 2019; Zheng et al., 2018). In our base DESC (Lopez-Rodriguez & Mikolajczyk, 2020), we adopt the approach from T^{2}Net (Zheng et al., 2018), where a network $G_{\mathcal{S}\to\mathcal{T}}$ translates the source image to the target domain without cycle consistency. T^{2}Net (Zheng et al., 2018) uses a least-squares adversarial term $\mathcal{L}_{GAN}$ (Mao et al., 2017) to produce examples $x_{\mathcal{S}\to\mathcal{T}}$ having a similar distribution to $x_{\mathcal{T}}$, and leverages the constraint imposed by $\mathcal{L}_D$ to ensure $x_{\mathcal{S}\to\mathcal{T}}$ is geometrically consistent with $x_{\mathcal{S}}$. The method also uses a L_1 identity loss $\mathcal{L}_{IDT} = \frac{1}{N}\|G_{\mathcal{S}\to\mathcal{T}}(x_{\mathcal{T}}) - x_{\mathcal{T}}\|_1$ to force $G_{\mathcal{S}\to\mathcal{T}}(x_{\mathcal{T}}) \approx x_{\mathcal{T}}$, i.e., $\mathcal{L}_{IDT}$ forces $G_{\mathcal{S}\to\mathcal{T}}$ to behave as an identity mapping for $x_{\mathcal{T}}$. In this follow-up work to DESC (Lopez-Rodriguez & Mikolajczyk, 2020), we also present results in Sect. 4.2 when using newer image translation techniques such as SharinGAN (PNVR et al., 2020) and S2R-DepthNet (Chen et al., 2021), which yield improved results compared to using the T^{2}Net-approach.

Smoothing We use for the target data the smoothing term $\mathcal{L}_{Sm}$ introduced in Monodepth (Godard et al., 2017), and successfully used in domain adaptation (Zheng et al., 2018;

Zhao et al., 2019) methods for depth estimation. The smoothing term encourages the predicted depth map to be locally smooth except in those areas where there are large gradients in the RGB image, as those regions are likely to have depth discontinuities. $\mathcal{L}_{Sm}$ is thus defined as:

$$\mathcal{L}_{Sm} = \sum_{s} \frac{1}{2^s N} \sum_{i,j} |\partial_x G_D(x_{\mathcal{T}})_s| e^{-\|\partial_x x_{\mathcal{T}i,j}\|} + |\partial_y G_D(x_{\mathcal{T}})_s| e^{-\|\partial_y x_{\mathcal{T}i,j}\|} \tag{5}$$

where i, j refer to pixel i, j, and ∂_x and ∂_y are the gradients in dimensions x and y. As Eq. (5) shows, the weight of $\mathcal{L}_{Sm}$ is reduced by 2^s for the higher resolution predicted depth maps.

Overall Loss Our final model is trained using the following loss

$$\mathcal{L} = \lambda_{\mathcal{S}}(\mathcal{L}_{\mathcal{D}} + \mathcal{L}_{Con,\mathcal{S}} + \mathcal{L}_{I,\mathcal{S}}) + \lambda_{\mathcal{T}}(\mathcal{L}_{Con,\mathcal{T}} + \mathcal{L}_{I,\mathcal{T}}) + \lambda_{Sm}\mathcal{L}_{Sm} + \lambda_{IDT}\mathcal{L}_{IDT} + \lambda_{GAN}\mathcal{L}_{GAN} \tag{6}$$

where $\lambda_{\mathcal{S}}, \lambda_{\mathcal{T}}, \lambda_{Sm}, \lambda_{IDT}, \lambda_{GAN}$ are hyperparameters to balance the different terms.

4 Experiments

We discuss the experimental setup before presenting our evaluation results.

Setup We use Pytorch 1.4 and an NVIDIA 1080 Ti GPU. We obtain the semantic annotations, in both $\mathcal{S}$ and $\mathcal{T}$, by using a ResNet-101 (He et al., 2016) panoptic segmentation model (Kirillov et al., 2019a) trained on COCO-Stuff (Lin et al., 2014; Caesar et al., 2018) from the *Detectron 2* library (Wu et al., 2019). We employ a VGG-based U-Net (Ronneberger et al., 2015) for G_D and G_S, and a ResNet-based model for $G_{\mathcal{S}\to\mathcal{T}}$. Both image translation and depth estimation architectures are the same as the architectures used in Zheng et al. (2018) and Zhao et al. (2019). Following Zhao et al. (2019), we set $\lambda_{\mathcal{S}} = 50$, $\lambda_{GAN} = 1$, $\lambda_{Sm} = 0.01$, and following Zheng et al. (2018) we set $\lambda_{IDT} = 100$. Similarly to the original implementation of Zhao et al. (2019), we first pretrain the networks to reach good performance in $\mathcal{S}$ before introducing the consistency terms, i.e., with $\lambda_{\mathcal{T}} = 0$. Afterwards, we freeze $G_{\mathcal{S}\to\mathcal{T}}$ to reduce the memory footprint, and we introduce the semantic consistency terms by setting $\lambda_{\mathcal{T}} = 1$ unless stated otherwise. The batch size is set to 4, with a 50/50 target and source data ratio, we use Adam (Kingma & Ba, 2015) with learning rate 10^{-4} and we train for 20000 iterations after pretraining. To obtain the edge map for G_S we use a Canny Edge detector (Canny, 1986). We randomly change the brightness, saturation and contrast of the images for data augmentation.

Virtual KITTI→KITTI We follow the same experimental settings as in Zheng et al. (2018) and Zhao et al. (2019). Both Virtual KITTI (Gaidon et al., 2016) and KITTI (Geiger et al., 2012) images are downscaled to 640 × 192, and following Zheng et al. (2018) we cap the Virtual KITTI (Gaidon et al., 2016) ground truth depth at 80 m.

Cityscapes→KITTI Cityscapes (Cordts et al., 2016) provides disparity maps computed using Semi-Global Matching (Hirschmuller, 2007). We use the official training set, consisting of 2975 images of size 2048 × 1024. We set the horizon line approximately in the center by cropping the upper part, resulting in images of 2048 × 964. We then take the 2048 × 614 center crop to have the same aspect ratio as in KITTI and rescale the images to 640 × 192. We use $\lambda_{\mathcal{T}} = 5$ for this experiment.

Evaluation on KITTI. We follow the same evaluation protocol, metrics and splits as in Eigen et al. (2014) for KITTI, using the evaluation code from Monodepth2 (Godard et al., 2019). The predictions are upscaled to match the ground truth size. The results are reported using median scaling as in past methods (Nath Kundu et al., 2018; Casser et al., 2019; Zhou et al., 2017), except when using stereo supervision in KITTI or in a semi-supervised regime. We provide results for both ground truth depth capped at 80 m and between 1 and 50 m as done in Zhao et al. (2019) and Zheng et al. (2018).

4.1 Results on Virtual Kitti→Kitti

Comparison with State-of-the-Art Table 1 compares the performance of DESC with the state-of-the-art Virtual KITTI→KITTI methods that do not use stereo nor video self-supervision in KITTI. DESC performs better than AdaDepth (Nath Kundu et al., 2018) and T^{2}Net (Zheng et al., 2018), with a Sq. Rel. error almost 24% lower than T^{2}Net. Compared to *S^{3}Net* (Cheng et al., 2020), which also uses semantic information during training, DESC performs better in most metrics than *S^{3}Net* but is outperformed by *S^{3}Net (Test Semantic)*, as the latter employs semantic maps during test time. The recent domain generalization method S2R-DepthNet (Chen et al., 2021), published after DESC (Lopez-Rodriguez & Mikolajczyk, 2020), achieves comparable results to DESC without using any KITTI data during training, especially in the 80m cap setting. In Sect. 4.2, we explore the combination of S2R-DepthNet and DESC, which improves notably the results of the original DESC. Figure 4 shows some predictions of our DESC method compared to T^{2}Net, which we build upon. DESC contains fewer high-error regions than T^{2}Net (Zheng et al., 2018) due to the guidance provided by G_S, as shown in the upper-right wall of

Table 1 Results for Virtual KITTI→KITTI in KITTI (Geiger et al., 2012) Eigen (Eigen et al., 2014) split

Method	Lower is better				Higher is better		
	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Cap 80 m							
AdaDepth (Nath Kundu et al., 2018)	0.214	1.932	7.157	0.295	0.665	0.882	0.950
T^{2}Net (Zheng et al., 2018)	0.173	1.396	6.041	0.251	0.757	0.916	0.966
S2R-DepthNet (Chen et al., 2021)	0.162	1.339	5.684	**0.232**	0.786	**0.934**	**0.974**
DESC	**0.156**	**1.067**	**5.628**	0.237	**0.787**	0.924	0.970
Cap 50 m							
AdaDepth (Nath Kundu et al., 2018)	0.203	1.734	6.251	0.284	0.687	0.899	0.958
T^{2}Net (Zheng et al., 2018)	0.165	1.034	4.501	0.235	0.772	0.927	0.972
S2R-DepthNet (Chen et al., 2021)	0.155	0.997	4.327	0.220	0.799	**0.941**	**0.978**
S^{3}Net (Cheng et al., 2020)	0.154	0.993	4.449	0.224	0.799	0.936	0.975
DESC	0.149	**0.819**	**4.172**	0.221	0.805	0.934	0.975
S^{3}Net (Test Semantic) (Cheng et al., 2020)	**0.145**	0.887	4.218	**0.215**	**0.813**	**0.941**	0.977

For a fair comparison, we use the official pretrained models given by *T^{2}Net* and *S2R-DepthNet* to recompute the results using median scaling. *S^{3}Net* is the best reported result in Cheng et al. (2020) not using any semantic maps at test time, whereas *S^{3}Net* (*Test Semantic*) uses semantic maps both at train and test time. S2R-DepthNet (Chen et al., 2021) is a domain generalization method that does not use any KITTI data during training Bold values refer to the best performance obtained per metric and category

the predictions in the first row. The geometry of the instances in our method tends to be complete, e.g., the cars of the second row and the larger car in the first row, which has large missing parts in the T^{2}Net prediction. The last row in Fig. 4 also shows that DESC produces less detailed regions due to the consistency term with G_S blurring the predictions and removing some fine structures.

Ablation Study Table 2 shows an ablation study of DESC. The result marked with $+Img$ correspond to T^{2}Net (Zheng et al., 2018) without the adversarial feature module, and with a lower smoothing weight λ_{Sm} as we use $\lambda_{Sm} = 0.01$ instead of the $\lambda_{Sm} = 0.1$ used for the T^{2}Net implementation shown in Table 1. The lower λ_{Sm} we use accounts for the better results of T^{2}Net in Table 1. We chose a smaller λ_{Sm} for our experiments because a larger λ_{Sm} blurs the predictions, leading to a worse result after enforcing consistency with G_S due to the loss of detail. However, when consistency with G_S is not applied, a larger λ_{Sm} is beneficial as shown by the improved results of *+Img.+Ins.* ($\lambda_{Sm} = 0.1$) compared to *+Img.+Ins.*. Both the instance-based pseudo-labelling and consistency with G_S modules bring an improvement as shown in *+Img.+Ins.* and *+Img.+Con.* compared to *+Img*. Using the consistency term when only edge maps are input into G_S improves most metrics as shown in *+Img.+Con.* (*only edges*), although it also shows that inputting the semantic map into G_S is largely beneficial. *DESC − Full* shows an improvement in all metrics, also compared to learning a single h per class in *DESC − Full* (*1 h per class*) as in Struct2Depth (Casser et al., 2019). For *DESC − Full* (*unknown* $f_{\mathcal{T}}$) we set $f_{\mathcal{T}}$ to half the actual value, obtaining comparable results to when using the correct value of $f_{\mathcal{T}}$, i.e., in *DESC − Full*. This result shows that ϕ in Eq. (2) automatically scales the instance size pseudo-labels to the correct range for unknown $f_{\mathcal{T}}$. Figure 5 shows some visual examples of the baseline with image translation (*Img*), of G_D trained with the two different losses we propose (*Img+Ins* and

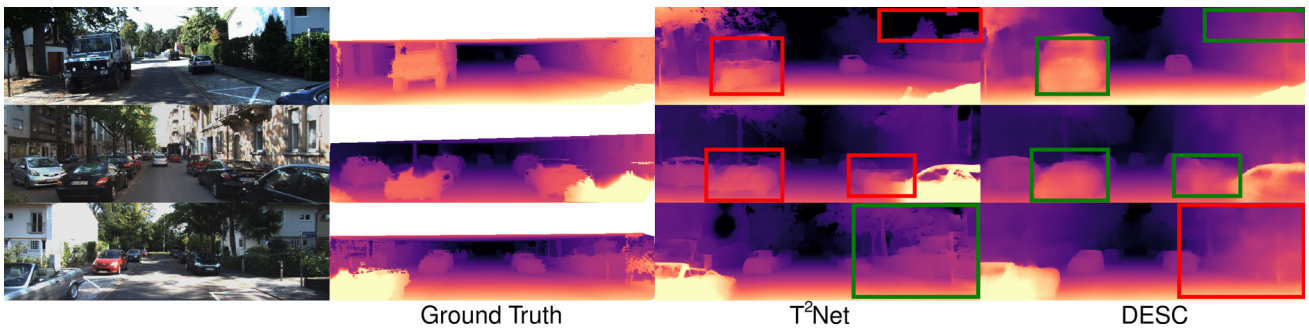

Fig. 4 Qualitative results in KITTI for models trained on Virtual KITTI→KITTI. Ground truth depth is linearly interpolated for visualization. Green bounding boxes refer to areas of the prediction more accurate compared to the corresponding red bounding boxes (Color figure online)

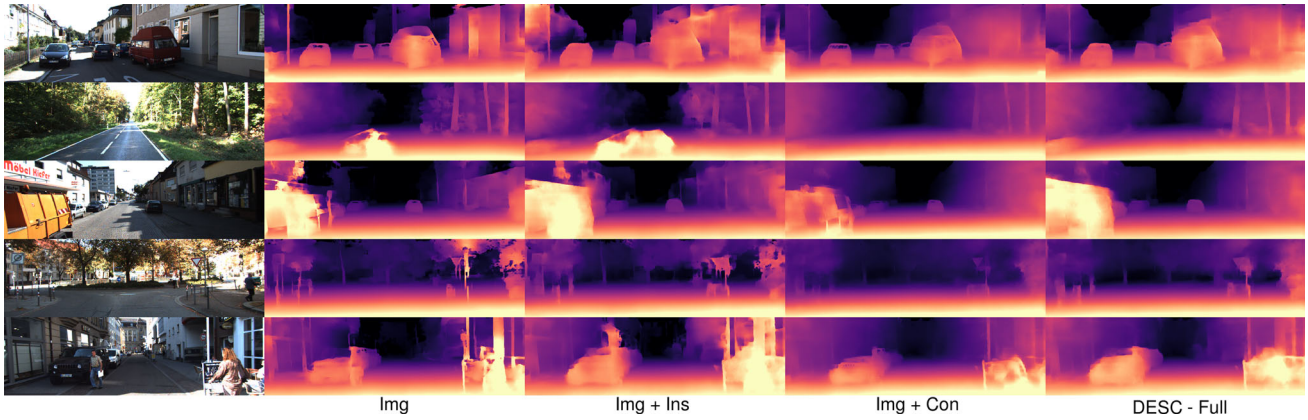

Fig. 5 Qualitative results of our ablation study. *Img* corresponds to our model trained with only the image translation module, *Img* + *Ins* combines the image translation module with the instance-based pseudo-labelling proposed in Sect. 3.1, *Img* + *Con* combines the image translation module with the consistency loss in Sect. 3.2, and *DESC* − *Full* is our complete pipeline

Img+Con), and of our full DESC pipeline. Figure 5 shows that adding the instance-based pseudo-labelling (*Img+Ins*) results in better completeness of the different instances compared to the T^{2}Net-based baseline (*Img*), which can be observed in e.g., the black car in the fifth row or the red van in the first row. Our consistency loss (*Img+Con*) improves in turn the overall structure of the scene, e.g., it corrects the missing depth values in the first-row wall or the errors on the road in the second row. We also observe how our consistency loss results in a loss of details as it tends to smooth the predictions. Finally, the full model (*DESC* − *Full*) combines the better scene structure and higher smoothness obtained when using the consistency loss, with the better instance completeness obtained with the instance-based pseudo-labelling loss.

Virtual KITTI 2 (Cabon et al., 2020) is an updated version of Virtual KITTI that replicates Virtual KITTI while improv-

Table 2 Ablation study of DESC for Virtual KITTI→KITTI in Eigen split (Eigen et al., 2014) capped at 80 m

Method	Lower is better				Higher is better		
	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Only source (Zhao et al., 2019)	0.223	2.205	7.055	0.305	0.672	0.872	0.945
+Img.	0.199	2.436	7.137	0.280	0.753	0.890	0.950
+Img. + Con. (only edges)	0.187	1.330	6.094	0.258	0.708	0.905	0.966
+Img. + Con.	0.173	1.235	5.776	0.244	0.748	0.919	0.969
+Img. + Ins.	0.171	1.332	5.818	0.250	0.771	0.918	0.966
+Img. + Ins. ($\lambda_{Sm} = 0.1$)	0.165	1.157	5.670	0.245	0.774	0.921	0.968
DESC−Full (1 h per class (Casser et al., 2019))	0.160	1.107	5.746	0.243	0.780	0.920	0.968
DESC−Full (unknown $f_{\mathcal{T}}$)	0.156	1.084	5.654	0.237	0.783	0.926	0.971
DESC−Full	0.156	1.067	5.628	0.237	0.787	0.924	0.970
DESC−Full (VKITTI2)	0.155	1.097	5.597	0.238	0.786	0.926	0.970
DESC−Full (R50)	0.160	1.207	6.034	0.248	0.777	0.918	0.965
DESC−Full (R50-ImageNet Pretr.)	**0.149**	**1.026**	**5.476**	**0.228**	**0.797**	**0.935**	**0.975**
EfficientPS Panoptic Maps	0.156	1.067	5.559	0.238	0.781	0.926	0.970
G_S− Synth	0.186	2.164	7.011	0.282	0.763	0.894	0.949
G_S− DESC	0.155	1.146	5.601	0.232	0.789	0.930	0.974
G_S− Synth + Stereo	0.136	1.206	5.598	0.235	0.822	0.932	0.969

Img. refers to using image translation, *Ins.* to using instance-height pseudo-labels (Sect. 3.1) and *Con.* to the consistency of predictions constraint (Sect. 3.2)

EfficientPS Panoptic Maps refers to our *DESC - Full* pipeline trained with panoptic maps from EfficientPS (Mohan and Valada, 2021) trained on Cityscapes (Cordts et al., 2016) instead of the Detectron2 (Wu et al., 2019) trained on COCO (Lin et al., 2014; Caesar et al., 2018) we use for the rest of the experiments. We also include the results obtained when evaluating the output of the network G_S when G_S is trained only with dense supervision from the source data (G_S - *Synth*), after our full pipeline (G_S - *DESC*), and when trained jointly with synthetic dense supervision and target-domain stereo data (G_S - *Synth* + *Stereo*)

Bold values refer to the best performance obtained per metric with our main G_D model

Table 3 Results on KITTI Eigen split (80 m cap) for methods using stereo data in KITTI

Method	Lower is better				Higher is better		
	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Virtual KITTI→ KITTI							
Source + Stereo	0.131	1.154	5.518	0.227	0.837	0.937	0.971
T^2Net (Zheng et al., 2018) + Stereo	0.126	1.114	5.429	0.223	0.839	0.938	0.971
GASDA (Zhao et al., 2019)	0.124	1.018	5.202	0.217	0.846	0.944	0.973
DESC + Stereo	0.119	**0.935**	**5.050**	0.217	0.843	0.942	0.974
SharinGAN (PNVR et al., 2020)	**0.116**	0.939	5.068	**0.203**	**0.850**	**0.948**	**0.978**
Only KITTI							
Monodepth2 (w/o pre.) (Godard et al., 2019)	0.130	1.144	5.485	0.232	0.831	0.932	0.968
Monodepth2 (ImageNet pre.) (Godard et al., 2019)	0.109	0.873	4.960	0.209	0.864	0.948	0.975

Due to an evaluation error in Zhao et al. (2019), results from GASDA are recomputed using the official pretrained models. We include one of the state-of-the-art stereo-trained methods *Monodepth2* (Godard et al., 2019)
Bold values refer to the best performance obtained per metric for the models leveraging both Virtual KITTI and KITTI data, which do not use a network pretrained on ImageNet

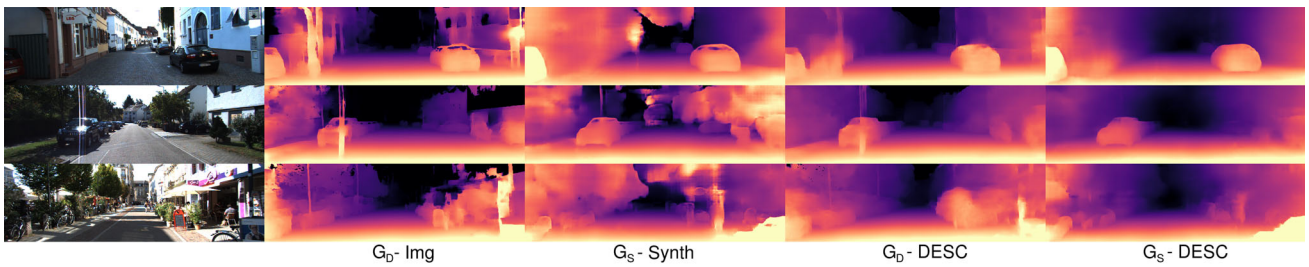

Fig. 6 Qualitative results of G_S and G_D before ($G_D - Img$ and $G_S - Synth$) and after ($G_D - DESC$ and $G_S - DESC$) consistency training

ing the visual quality of the synthetic images. We include in Table 2 the performance of DESC when using Virtual KITTI 2 as our source data. Despite the higher-quality synthetic images, the results of DESC are comparable when using either Virtual KITTI or Virtual KITTI 2, which we attribute to the image translation module outputting similar source to target translations to those when using Virtual KITTI.

ImageNet Pretrained G_D. Our G_D network is a randomly initialized VGG-based model, which was also employed in past domain adaptation works (Zheng et al., 2018; Zhao et al., 2019). However, some state-of-the-art depth estimation works (Godard et al., 2019) use a ResNet-based network initialized with ImageNet weights, as transfer learning has proven to be beneficial for depth estimation (Alhashim & Wonka, 2018). In that direction, we run our DESC approach using for G_D the ResNet50-based network (He et al., 2016) with an ImageNet pretrained encoder given in the Monodepth2 (Godard et al., 2019) implementation. Compared to our original VGG-based U-Net, we obtain lower performance with the randomly initialized ResNet-50 as shown in $DESC - Full(R50)$ in Table 2. However, when using an ImageNet initialized encoder, line $DESC - Full$ (*R50-ImageNet Pretr.*) in Table 2, the absolute relative error is improved by 4.5% compared to $DESC - Full$, which highlights the importance of ImageNet pretraining also in a domain adaptation setting as we obtain faster training and better performance.

Panoptic Model For the semantic segmentation map used in G_S we leveraged a COCO-trained model using the *Detectron 2* (Wu et al., 2019) library. However, COCO includes both high diversity images and classes that are not relevant to the driving data in KITTI/Virtual KITTI. For that reason, we aim to test if using the panoptic predictions from a model trained in a more similar domain, in this case the state-of-the-art method EfficientPS (Mohan & Valada, 2021) trained on Cityscapes (Cordts et al., 2016), has an impact on the performance. Table 2 shows that the results are comparable for most metrics when using either Detectron2, line $DESC - Full$, or EfficientPS, in the line *EfficientPS Panoptic Maps*. This similarity of results suggests that using a panoptic segmentation model trained on a more general dataset, i.e., COCO, seems to not degrade the performance compared to selecting a more domain-specific panoptic model.

Performance of G_S. Table 2 includes the results of G_S after being trained only with Virtual KITTI data (G_S- *Synth*), and when combining Virtual KITTI dense supervision and KITTI stereo supervision (G_S- *Synth+Stereo*). Even though

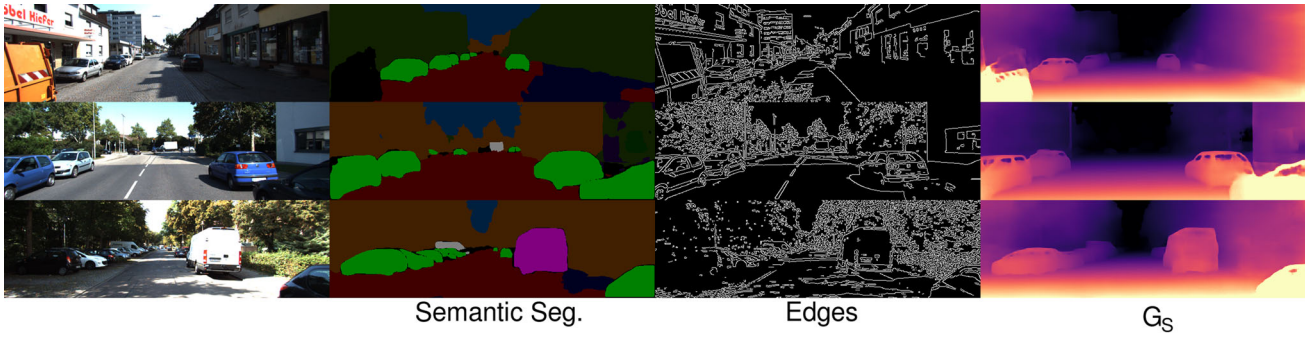

Fig. 7 Qualitative results of G_S, which takes as input the concatenation of the semantic and edge map

G_S only leverages edges and semantic information, the performance of G_S with stereo supervision is comparable to the main network G_D with stereo supervision (Table 3, line *Source+Stereo*). Furthermore, we argue that the improved results of G_D after applying semantic consistency (Table 2, line *+Img.+Con.*) compared to only using image translation (Table 2, line *+Img*) are not due to a distillation process, i.e., due to G_S having a higher accuracy and transferring its performance to G_D after source data pretraining. Instead, the lower performance of G_S- *Synth* compared to both G_D after applying semantic consistency (*+Img.+Con.*) and G_S after full training (G_S- *DESC*), suggests that consistency between predictions from different modalities constrains the learning process and is the reason for the accuracy increase, as both G_D and G_S models benefit from the consistency loss term. On that note, Fig. 6 shows that before applying consistency training (second and third column), both models differ in the artifacts and errors shown due to the different input modalities used. For example, G_D shows an incorrect prediction on the right side wall of the three given examples and mistakes an illumination effect as a depth change in the left-side car in the second row. However, as G_S uses semantic and edge maps as inputs, it provides a smoother prediction on those walls and is not as affected by illumination changes, but presents other types of artifacts. Our consistency loss forces the models to agree and corrects those mistakes in both G_S and G_D (fourth and fifth column), which leads to a better scene structure but also to a loss of details. Furthermore, in Fig. 7 we show some examples of both inputs and predictions of our G_S module. The module is highly guided by the semantic segmentation mask, where errors in the prediction (e.g., missing part of the left van in the top image and non-straight edges in the bottom right car in the middle row) translate to errors in the prediction. The edge map does help recover some details, e.g., the car windows, but overall the resulting predictions are smooth (e.g., foliage in the bottom row prediction) due to the difficulty of predicting high-frequency changes from edges and semantic maps.

Performance of G_h. We now test the performance of our instance height pseudo-labelling approach. Table 4 shows the errors when directly evaluating the pseudo-label obtained combining Eq. (1) and our fully-trained G_h. We also include the performance when using in Eq. (1) the optimal per-class h (*Opt. Class h*), which is obtained by finding the per-class value that minimizes the *Abs Rel* metric in the test set, and acts as an upper bound of the performance possible to obtain using a single h per class. Our trained G_h is capable of outperforming the optimal per-class h in most metrics, which validates the choice of an instance-based height prediction method. We also show that introducing our instance-based pseudo-labels

Table 4 Results on KITTI Eigen split (cap 80 m) for three common classes (*car*, *person*, *bike*) using the same *Detectron 2* library we leverage during training

Method	Lower is better			
	Abs Rel	Sq Rel	RMSE	RMSE log
Car				
Opt. Class h	0.205	1.665	6.526	0.395
G_h	0.177	1.571	6.628	0.403
G_D - Img + Con.	0.207	1.598	**6.004**	**0.324**
G_D - DESC Full	**0.164**	**1.295**	6.240	0.355
Person				
Opt. Class h	0.204	2.112	6.574	0.373
G_h	**0.202**	**1.668**	6.384	**0.362**
G_D - Img + Con.	0.588	6.307	8.566	0.520
G_D - DESC Full	0.273	1.952	**6.118**	**0.362**
Bike				
Opt. Class h	0.161	0.535	2.548	0.227
G_h	**0.146**	0.496	2.775	0.207
G_D - Img + Con.	0.170	0.493	**2.483**	**0.202**
G_D - DESC Full	0.153	**0.451**	2.612	0.211

The results are averaged over all pixels with valid ground-truth, and for these results no median scaling is applied. *Opt. Class h* is the value that achieves the lowest error on the KITTI Eigen split test set, which is 1.50 m for car, 1.65 m for person and 1.15 m for bike

Bold values refer to the best performance obtained per metric and semantic class

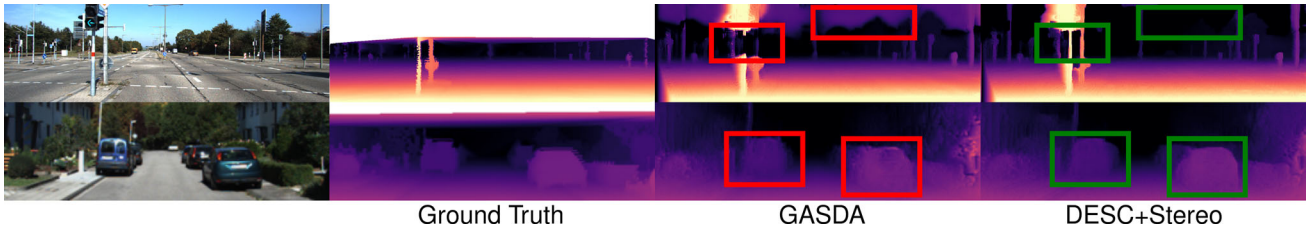

Fig. 8 Qualitative results in KITTI for models trained on Virtual KITTI→KITTI with stereo supervision in KITTI. Bottom row corresponds to a center crop of the original image

in our G_D training (G_D − *DESC Full*) improves the performance in the classes shown in Table 4 compared to when not using our pseudo-labelling approach (G_D-*Img* + *Con*), especially in the *Abs Rel* and *Sq Rel* metrics.

Stereo Supervision Although DESC focuses on the setting where no self-supervision is used in $\mathcal{T}$, our approach can also bring an improvement in such a scenario. We train DESC adding stereo supervision in KITTI by adding the same multiple-scale pixel-wise reconstruction method as in GASDA (Zhao et al., 2019) with the same loss weight of $\lambda_{St} = 50$. To account for the introduced supervision in $\mathcal{T}$, we increase $\lambda_{\mathcal{T}} = 5$ and the number of training iterations to 100,000. Table 3 shows that, compared to T^2*Net+Stereo*, our method with stereo supervision, *DESC + Stereo*, achieves better results in all metrics and also outperforms GASDA (Zhao et al., 2019) in most metrics. GASDA is a domain adaptation method tailored for stereo supervision that uses two depth estimation networks and an image-translation network during inference. The recent stereo-focused domain adaptation method SharinGAN (PNVR et al., 2020), which is concurrent to the original DESC (Lopez-Rodriguez & Mikolajczyk, 2020), performs better in most metrics than *DESC + Stereo* due to the improved image transfer strategy used, although SharinGAN also increases the computational cost at test time due to using an extra network. We also report better performance than the state-of-the-art for stereo supervision, Monodepth2 (Godard et al., 2019) without ImageNet (Deng et al., 2009) pretraining in *Monodepth2 (w/o pre.)*. However, ImageNet pretraining has a large effect on the accuracy of Monodepth2, shown in *Monodepth2 (ImageNet pre.)*, achieving better results than our method. Figure 8 shows predictions for domain adaptation methods using stereo supervision in KITTI. Compared to GASDA, we observe a better recovery of fine structures, shown in the pole of the first row of Fig. 8, and better predictions of further object instances, shown in the bottom row. DESC also predicts a better depth for the sky, as shown in the first row of Fig. 8.

Hyperparameter Selection The weights associated with the image translation process were chosen following T^{2}Net values, however we still need to tune the semantic consistency weight $\lambda_{\mathcal{T}}$. In that direction, we include in Table 5 the effects of varying $\lambda_{\mathcal{T}}$. Multiplying the original $\lambda_{\mathcal{T}} = 1$ by 5, i.e., line $\lambda_{\mathcal{T}} = 5$, the absolute relative error degrades by 4%, although it still achieves better performance than T^{2}Net (Table 1). Table 5 also shows the results when varying the number of training iterations when applying the last semantic consistency step of training. Even though modifying the training iterations impact the performance and may lead to overfitting for larger training iteration values, the error variation is less pronounced compared to modifying $\lambda_{\mathcal{T}}$. In practice, most unsupervised domain adaptation methods use quantitative performance in the target domain to tune to some level the hyperparameters or the model. Further research needs to develop reliable methods to avoid any assumption of target domain ground truth for hyperparameter or model selection (You et al., 2019).

Table 5 Results on KITTI Eigen split (cap 80 m) when varying either the weight of the semantic consistency loss $\lambda_{\mathcal{T}}$, and the number of training iterations for the last step of training

Method	Lower is better			
	Abs Rel	Sq Rel	RMSE	RMSE log
Variation of $\lambda_{\mathcal{T}}$				
$\lambda_{\mathcal{T}} = 1.0$	**0.156**	**1.067**	**5.628**	**0.237**
$\lambda_{\mathcal{T}} = 2.0$	0.158	1.081	5.816	0.243
$\lambda_{\mathcal{T}} = 5.0$	0.162	1.143	6.044	0.249
Train. It. ($\lambda_{\mathcal{T}} = 1$)				
10000	0.158	1.105	**5.557**	**0.236**
20000	**0.156**	**1.067**	5.628	0.237
40000	0.157	1.077	5.661	0.239
80000	0.159	1.099	5.784	0.244

Default values used in DESC are $\lambda_{\mathcal{T}} = 1.0$ and 20,000 training iterations

Bold values refer to the best performance obtained per metric and category

Distribution of Errors Following Gurram et al. (2021), we now analyze the obtained error of our methods *DESC* and *DESC+Stereo* for different depth ranges and semantic classes. As DESC builds upon T^{2}Net, we also include it in the analysis to better understand the performance improvement given by DESC. Figure 9 shows the distribution of errors depending on the ground truth value. KITTI concen-

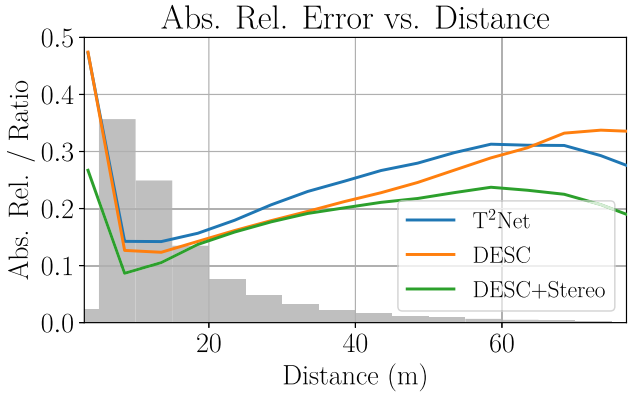

Fig. 9 Absolute relative error versus ground truth distance for three different methods in the KITTI Eigen test split. We also include a histogram of the ground truth distribution. Left axis shows both absolute relative error (line plot) and ratio of ground truth values (histogram)

trates most of the ground truth values around 5-20 meters, where the represented methods also present their lower absolute relative error. Our method, DESC, consistently performs better than T^{2}Net for depth values under $\approx$ 65m, however the performance of DESC drops for depth values close to 80m. Adding stereo information to DESC improves the performance largely for closer depth values (where there are large disparity shifts) and for larger depth values, whereas it achieves similar performance to regular DESC for mid-range values. Figure 10 shows the performance averaged over all the different semantic classes (we use the EfficientPS semantic maps trained on Cityscapes for evaluation). DESC performs better than T^{2}Net in most of the classes, especially for the detected object instances (e.g., car, train or traffic light) due to the semantic consistency introduced in DESC, which is also related to the better completeness of object instances shown in Fig. 4.

4.2 Improved Image Transfer Strategies

The image transfer approach used in DESC (Lopez-Rodriguez & Mikolajczyk, 2020) is based upon T^{2}Net (Zheng et al., 2018). However, recent domain adaptation and generalization works outperformed the image transfer method in T^{2}Net. We now aim to substitute the T^{2}Net strategy used in DESC by these improved image transfer strategies. Specifically, we include in DESC the methods presented in SharinGAN (PNVR et al., 2020) and the domain generalization method S2R-DepthNet (Chen et al., 2021), which were discussed in Sect. 2.3. To apply SharinGAN to DESC we use the image transfer network pretrained on Virtual KITTI→KITTI from the official SharinGAN code and keep it frozen during training. For S2R-DepthNet, we take the official pretrained models, which are trained on Virtual KITTI, and further finetune both the depth network and the attention network using our semantic consistency modules. For both SharinGAN and S2R-DepthNet approaches, we translate both target and source data to an intermediate domain before feeding the images to G_D.

Quantitative Results Table 6 shows the results for all the three image transfer approaches used with our DESC method. Both SharinGAN and S2R-DepthNet further improved the performance of DESC. Using SharinGAN instead of a T^{2}Net approach in DESC decreases the absolute relative error in cap 80m by 2%, whereas using S2R-DepthNet has a larger impact on the DESC performance, decreasing the absolute relative error by 7%. *DESC-S2R-DepthNet* outperforms in all metrics both the base S2R-DepthNet and base DESC results given in Table 1. The increased performance achieved when combining DESC with either method also shows the wide applicability of DESC to other image translation methods.

Qualitative Results Figure 11 shows examples of image translations for our trained T^{2}Net approach, the SharinGAN model and the depth structure output by S2R-DepthNet. T^{2}Net produces a stronger shift in the Virtual KITTI images compared to SharinGAN, resembling more the illumination and textures present in the real KITTI at the cost of introducing artifacts (e.g., hallucinated trees). Furthermore, Figs. 11 and 12 show that the SharinGAN translations for both Virtual KITTI and KITTI images are quite close to the input image, suggesting that non-aggressive changes are enough to achieve good performance. S2R-DepthNet produces images quite different to those from either T^{2}Net or SharinGAN.

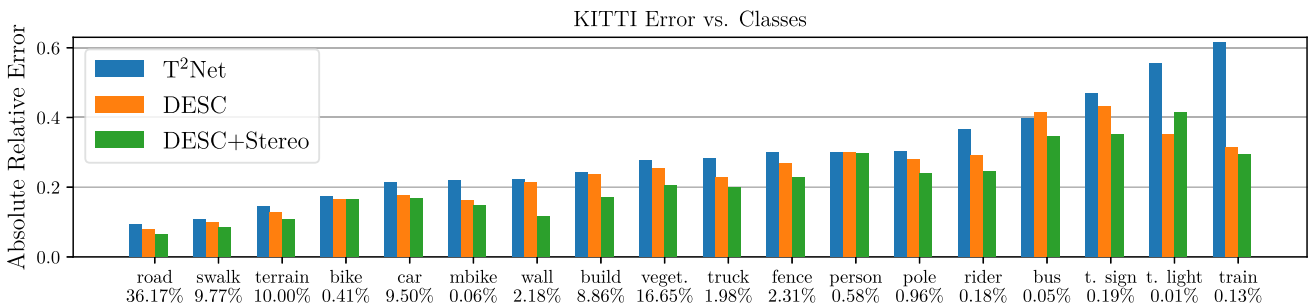

Fig. 10 Distribution of absolute relative error averaged over all pixels for a specific detected class over the KITTI Eigen test split (cap 80 m) for models three different methods trained on Virtual KITTI→KITTI. The percentage of ground truth depth corresponding to each class is given below each class

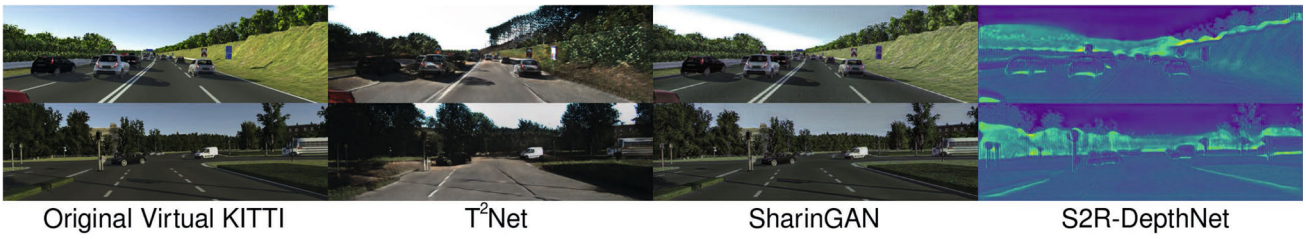

Fig. 11 Qualitative results for three different image transfer strategies. Virtual KITTI images are transferred to either a style matching KITTI images using a T^{2}Net-based approach or to an intermediate shared domain with the transferred KITTI images using either a SharinGAN (PNVR et al., 2020) or S2R-DepthNet (Chen et al., 2021) approach. The original single-channel S2R-DepthNet images are mapped to RGB using a colormap and logarithmic mapping

Fig. 12 Transfer of KITTI images to the intermediate shared domain employed by SharinGAN (PNVR et al., 2020) and S2R-DepthNet (Chen et al., 2021). The original single-channel S2R-DepthNet images are mapped to RGB using a colormap and logarithmic mapping

Table 6 Results of DESC on the KITTI Eigen test split when combined with three different image transfer modules

Method	Lower is better				Higher is better		
	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Cap 80 m							
DESC − T^{2}Net (Zheng et al., 2018)	0.156	1.067	5.628	0.237	0.787	0.924	0.970
DESC − SharinGAN (PNVR et al., 2020)	0.153	1.057	5.641	0.236	0.789	0.926	0.970
DESC − S2R − DepthNet (Chen et al., 2021)	**0.145**	**0.986**	**5.368**	**0.225**	**0.806**	**0.938**	**0.976**
Cap 50 m							
DESC − T^{2}Net (Zheng et al., 2018)	0.149	0.819	4.172	0.221	0.805	0.934	0.975
DESC − SharinGAN (PNVR et al., 2020)	0.146	0.804	4.123	0.219	0.807	0.937	0.975
DESC − S2R − DepthNet (Chen et al., 2021)	**0.139**	**0.742**	**3.971**	**0.211**	**0.821**	**0.947**	**0.980**

Bold values refer to the best performance obtained per metric and category

S2R-DepthNet mostly removes texture and illumination cues (e.g., no shadows in Fig. 11 cars) and only keeps the structural edges needed for depth prediction. Hence, S2R-DepthNet maps the input RGB images to a lower-gap intermediate domain, as shown in Figs. 11 and 12 where the resulting S2R-DepthNet images from KITTI and Virtual KITTI are quite closer in appearance compared to the original RGB images.

Computational Complexity We use for our experiments a single NVIDIA 1080 Ti. Our base DESC only adds computational cost during training, hence the inference speed depends on the depth prediction network used. The U-Net employed in DESC for G_D, also used in T^{2}Net and GASDA, is capable of an inference of 43 imgs/s with a resolution of 640 × 192, assuming a batch size of 1. However, using the presented improved image-transfer strategies reduces the inference speed, as both SharinGAN and S2R-DepthNet approaches need extra networks at test time. In the case of using a *DESC-SharinGAN* approach, the inference speed decreases to 23 imgs/s, whereas with *DESC-S2R-DepthNet* we achieve a speed of 9 imgs/s. The total training time for our original DESC is approximately 2 days which accounts for the three training steps (i.e., pretraining of G_D, pretraining of G_S and joint training) and assumes the panoptic predictions are precomputed.

4.3 Evaluation in Additional Settings

Make3D (Saxena et al., 2008) is used to test the generalization capabilities of our DESC model trained in the Virtual KITTI→ KITTI scenario. The ground truth in Make3D is of low quality and low resolution, as shown in the examples in Fig. 13, hence the results provide only rough guidance of the generalization ability of the model. We use the evaluation protocol and code given by another domain adaptation method, SharinGAN (PNVR et al., 2020), for a fair comparison. We include in Table 7 the results of both our base DESC and our DESC with ImageNet pretraining. We also report

Table 7 Results on Make3D (70 m cap) (Saxena et al., 2008) using the same central image crop as PNVR et al. (2020)

Method	Lower is better		
	Abs Rel	Sq Rel	RMSE
No median scaling			
S2R-DepthNet	0.490	10.676	10.889
T^2Net	0.508	6.589	8.935
GASDA	0.403	6.709	10.424
SharinGAN	**0.377**	4.900	8.388
DESC	0.424	5.563	8.571
DESC (R50-ImageNet pretr.)	0.386	**3.943**	**8.104**
Median scaling			
S2R-DepthNet	0.485	10.547	10.833
T^2Net	0.420	7.477	9.992
GASDA	0.377	6.323	9.097
SharinGAN	0.322	3.744	7.812
DESC	0.335	3.772	8.030
DESC (R50-ImageNet pretr.)	**0.293**	**2.755**	**7.510**

All of the methods have been trained on Virtual KITTI→KITTI, except for S2R-DepthNet, which is only trained on Virtual KITTI. For non-median scaled results of SharinGAN, GASDA and T^2Net, we report the results given in PNVR et al. (2020), and we compute the median scaled results using the evaluation code given in PNVR et al. (2020) along with the official pretrained models given by each method
Bold values refer to the best performance obtained per metric and category

the results on Make3D both with and without median scaling, as median scaling greatly improves the results due to the different camera intrinsics and image resolution in Make3D affecting the scale of the predictions. Table 7 shows that our method performs comparatively well in Make3D, and the only domain adaptation method that obtains similar results is SharinGAN (PNVR et al., 2020), which contrary to DESC uses stereo information from the real domain during training. Compared to T^2Net (Zheng et al., 2018) and S2R-DepthNet, the other methods in Table 7 that do not use any real-domain stereo supervision during training, *DESC* achieves better performance by a wide margin. Figure 13 shows some qualitative results on Make3D, where we observe that the predictions of both DESC and T^2Net contain large areas of error, highlighting the need for methods capable of better generalization. The top row corresponds to an example where both methods fail to predict a satisfactory depth for the building, showing these generalization issues. The bottom row shows how our method, although quantitatively behaves noticeably better than T^2Net, produces blurrier predictions in Make3D as a consequence of the consistency loss with G_S used during training.

Semi-supervised Setting Past work (Zhao et al., 2020; Nath Kundu et al., 2018; Chen et al., 2021) has tackled a semi-supervised approach assuming access to 1000 KITTI images,

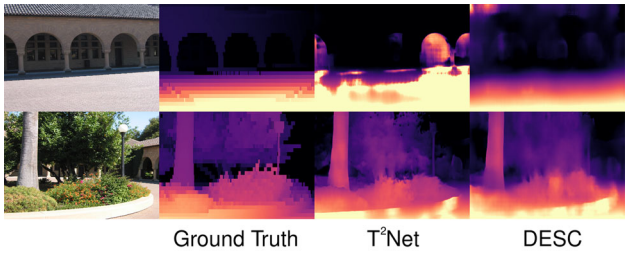

Fig. 13 Qualitative results in Make3D for T^2Net and DESC

which we now investigate. We use the same 1000 labelled KITTI frames in ARC (Zhao et al., 2020) as our annotated data. We finetune our final DESC model with the labelled real images following the same loss given in Eq. (6) with the addition of the KITTI ground truth supervision loss. For the target data supervision, as the ground truth is sparse, we upscale the feature maps instead of downscaling the ground truth to leverage all of the available sparse depth values. Table 8 shows that we obtain better results for *DESC - Only image translation*, which is a T^2Net without feature adaptation, compared to those reported in Zhao et al. (2020), which could be partially due to using a different implementation for the target loss. Table 8 also shows that DESC outperforms all of the past domain adaptation methods, but obtains lower performance than the domain generalization method S2R-DepthNet, which can be quickly adapted to new domains using few examples.

Evaluation on KITTI Stereo KITTI Stereo 2015 (Menze & Geiger, 2015) provides images annotated in a process combining (1) static background retrieval via egomotion compensation and (2) fitting of CAD models to account for dynamic objects. The result is a denser ground truth compared to the LiDAR depth annotations provided in KITTI, especially in the cars. DESC, which uses detected instances to generate depth pseudo-labels, benefits from evaluating in images with denser annotation in the vehicles, as shown in Table 9 in the larger accuracy gap between *DESC* and T^2*Net*, and also between *DESC* and *S2R-DepthNet*, which obtained comparable results on the Eigen split given in Table 1. Comparing stereo-trained methods we find a similar trend, there is a larger gap in performance between *DESC + Stereo* and *GASDA*, and *DESC + Stereo* also outperforms *SharinGAN* contrary to the results in Table 3. Furthermore, *DESC + Stereo* achieves either better (Sq Rel, RMSE) or equal (RMSE log) squared metrics results than the state-of-the-art *Monodepth2 (ImageNet pre.)* without pretraining G_D in ImageNet.

Cityscapes→ KITTI Table 10 shows the results for this benchmark. We improve upon T^2Net for all metrics, with a 13.9% lower absolute relative error. Most of the accuracy improvement comes from the consistency term as shown in *DESC (Img.+Con.)* and *DESC (Full, ϕ learnt)*. Due to the

Table 8 Results on the KITTI Eigen test split (cap 80 m) when using a semi-supervised setting with 1000 labelled KITTI images

Method	Lower is better				Higher is better		
	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
AdaDepthS (Nath Kundu et al., 2018)	0.167	1.257	5.578	0.237	0.771	0.922	0.971
Real + Syn	0.152	0.988	4.751	0.257	0.784	0.918	0.966
T^2Net	0.151	0.993	4.693	0.253	0.791	0.914	0.966
ARC	0.143	0.927	4.679	0.246	0.798	0.922	0.968
DESC - Only Img. Trans.	0.132	0.995	5.085	0.215	0.824	0.937	0.976
DESC	0.128	0.924	4.984	0.210	0.829	0.940	0.977
S2R-DepthNet	**0.116**	**0.766**	**4.409**	**0.185**	**0.858**	**0.955**	**0.984**

No median scaling performed during evaluation for this experiment. Results for *T^2Net*, *Real + Syn* and *ARC* taken from Zhao et al. (2020)
Bold values refer to the best performance obtained per metric

Table 9 Results on the KITTI 2015 stereo 200 training set disparity images (Menze & Geiger, 2015; Geiger et al., 2012)

Method	Lower is better				Higher is better		
	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Virtual KITTI→KITTI							
T^2Net (Zheng et al., 2018)	0.151	1.535	6.177	0.224	0.817	0.935	0.975
S2R-DepthNet (Chen et al., 2021)	0.142	1.371	5.737	0.207	0.835	0.949	0.981
DESC	0.120	0.968	5.597	0.206	0.839	0.937	0.977
GASDA (Zhao et al., 2019)	0.095	1.068	5.015	0.168	0.906	0.966	0.986
SharinGAN (PNVR et al., 2020)	0.092	0.903	4.611	0.159	0.906	**0.968**	**0.987**
DESC + Stereo	**0.085**	**0.781**	**4.490**	**0.158**	**0.909**	0.967	0.986
Only KITTI							
Monodepth2 (w/o pre.) (Godard et al., 2019)	0.096	1.163	5.161	0.179	0.898	0.959	0.981
Monodepth2 (ImageNet pre.) (Godard et al., 2019)	0.082	0.908	4.698	0.158	0.919	0.970	0.986

We include *Monodepth2* (Godard et al., 2019), the state-of-the-art stereo method trained only in KITTI. Results for non-stereo trained methods (*T^2Net*, *S2R-DepthNet* and *DESC*) are reported with median scaling
Bold values refer to the best performance obtained per metric for the models leveraging both Virtual KITTI and KITTI data, which do not use a network pretrained on ImageNet

camera difference between the datasets, the learnable scalar ϕ is necessary for good performance, as shown for fixed $\phi = 1$ in *DESC (Full, $\phi = 1$)*. Struct2Depth (Casser et al., 2019) also uses precomputed semantic annotations to improve its self-supervised video learning, although Struct2Depth is not a domain adaptation method as it only trains with Cityscapes (Cordts et al., 2016) data, i.e., it does not use KITTI for training. Struct2Depth also uses a different crop for Cityscapes. Table 10 shows that we achieve better accuracy than *Struct2Depth (M+R)*, which uses three frames at test time for refinement, whereas we only need a single image for inference.

4.4 Limitations

Due to the consistency term with G_S, our method shows some loss of detail in fine structures compared to T^2Net (Zheng et al., 2018), as shown in the last row of Fig. 4 or in Fig. 13, which could also limit the achievable upper-bound performance in settings with real data supervision, such as self-supervision or semi-supervised settings. Additionally, DESC is more computationally demanding during training than T^2Net due to the added G_S. The depth predicted by G_S also relies on the quality of the computed semantic data, hence in settings where the extracted annotations are of low quality the performance of the method may degrade. Furthermore, the instance-based pseudo-labelling predicts a height that assumes that the object is in an upright position, thus some rotations of the camera poses or objects could degrade the performance of that module.

5 Conclusion

We proposed a method that leverages semantic annotations to improve the performance of a depth estimation model in a domain adaptation setting. We used the relationship between instance size and depth to provide pseudo-labels in the tar-

Table 10 Cityscapes→KITTI results, evaluated in KITTI (Geiger et al., 2012) Eigen split (80 m cap)

Method	Lower is better				Higher is better		
	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Only Cityscapes							
Source Baseline	0.189	1.717	6.478	0.257	0.740	0.919	0.968
Struct2Depth (M) (Casser et al., 2019)	0.188	1.354	6.317	0.264	0.714	0.905	0.967
Struct2Depth (M+R) (Casser et al., 2019)	0.153	1.109	5.557	0.227	0.796	0.934	0.975
Cityscapes→ KITTI							
T^2Net (Zheng et al., 2018)	0.173	1.335	5.640	0.242	0.773	0.930	0.970
DESC (Img.+Ins.)	0.174	1.480	5.920	0.240	0.782	0.931	0.971
DESC (Img.+Con.)	0.150	0.981	5.359	**0.222**	0.805	0.938	**0.976**
DESC (Full, $\phi = 1$)	0.169	1.142	5.936	0.261	0.741	0.919	0.967
DESC (Full, ϕ learnt)	**0.149**	**0.967**	**5.236**	0.223	**0.810**	**0.940**	**0.976**

Struct2Depth (M+R) (Casser et al., 2019) uses three consecutive frames for refinement
Bold values refer to the best performance obtained per metric

get domain. A segmentation map and an edge map were input to a second network, whose prediction was forced to be consistent with the prediction of the main network. These additions led to higher accuracy in the setting where no self-supervision is available in the real data. In the Virtual KITTI to KITTI benchmark we outperform all of the other methods that do not use KITTI video or stereo supervision, and when employing a more advanced image strategy, we also outperform a method using semantic labels at test time. As we use automatically extracted semantic annotations, our method can be easily added to current approaches to improve their accuracy in a domain adaptation setting, as shown in the improvement achieved with stereo self-supervision or the multiple image-transfer strategies we successfully test. As future work, approaches aiming to reduce the detail loss due to the enforced consistency of predictions could improve the method.

References

Alhashim, I., & Wonka, P. (2018). *High quality monocular depth estimation via transfer learning*. arXiv e-prints pp arXiv-1812

Atapour-Abarghouei, A., & Breckon, T. P. (2018). Real-time monocular depth estimation using synthetic data with domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 1–8).

Atapour-Abarghouei, A., & Breckon, T. P. (2019). Veritatem dies aperit-temporally consistent depth prediction enabled by a multi-task geometric and semantic scene understanding approach. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 3373–3384)

Cabon, Y., Murray, N., & Humenberger, M. (2020). *Virtual KITTI 2*. arXiv e-prints arXiv:2001.10773

Caesar, H., Uijlings, J., & Ferrari, V. (2018). Coco-stuff: Thing and stuff classes in context. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 1209–1218).

Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 6,* 679–698.

Casser, V., Pirk, S., Mahjourian, R., & Angelova, A. (2019). Depth prediction without the sensors: Leveraging structure for unsupervised learning from monocular videos. In *Thirty-Third AAAI Conference on Artificial Intelligence (AAAI-19)*.

Chen, C., Xie, W., Huang, W., Rong, Y., Ding, X., Huang, Y., Xu, T., & Huang, J. (2019a). Progressive feature alignment for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 627–636).

Chen, P. Y., Liu, A. H., Liu, Y. C., & Wang, Y. C. F. (2019b). Towards scene understanding: Unsupervised monocular depth estimation with semantic-aware representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 2624–2632)

Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. (2020). A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning, PMLR* (pp. 1597–1607).

Chen, X., Wang, Y., Chen, X., & Zeng, W. (2021). S2r-depthnet: Learning a generalizable depth-specific structural representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 3034–3043).

Chen, Y., Li, W., Chen, X., Gool, L. V. (2019c). Learning semantic segmentation from synthetic data: A geometrically guided input-output adaptation approach. In *Proceedings of the IEEE/CVF*

Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 1841–1850).

Chen, Y. C., Lin, Y. Y., Yang, M. H., & Huang, J. B. (2019d). Crdoco: Pixel-level domain transfer with cross-domain consistency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 1791–1800).

Cheng, B., Saggu, I. S., Shah, R., Bansal, G., & Bharadia, D. (2020). S3net: Semantic-aware self-supervised depth estimation with monocular videos and synthetic data. In *Proceedings of the European Conference on Computer Vision (ECCV)* (pp. 52–69). Springer.

Choi, J., Jung, D., Lee, D., & Kim, C. (2020). Safenet: Self-supervised monocular depth estimation with semantic-aware feature extraction. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., & Schiele, B. (2016). The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 3213–3223).

Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., & Fei-Fei, L. (2009). ImageNet: A Large-Scale Hierarchical Image Database. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Dijk, T., & Croon, G. (2019). How do neural networks see depth in single images? In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 2183–2191).

Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., & Koltun, V. (2017). Carla: An open urban driving simulator. In *Proceedings of the Conference on Robot Learning (CoRL)* (pp. 1–16).

Eigen, D., Puhrsch, C., & Fergus, R. (2014). Depth map prediction from a single image using a multi-scale deep network. In *Advances in Neural Information Processing Systems (NeurIPS)* (pp. 2366–2374).

French, G., Mackiewicz, M., & Fisher, M. (2018). Self-ensembling for visual domain adaptation. In *Proceedings of the International Conference on Learning Representations (ICLR)* (p. 6).

Fu, H., Gong, M., Wang, C., Batmanghelich, K., & Tao, D. (2018). Deep Ordinal Regression Network for Monocular Depth Estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Gaidon, A., Wang, Q., Cabon, Y., & Vig, E. (2016). Virtual worlds as proxy for multi-object tracking analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 4340–4349).

Ganin, Y., & Lempitsky V. (2015). Unsupervised domain adaptation by backpropagation. In *Proceedings of the International Conference on Machine Learning (ICML)* (pp. 1180–1189).

Garg, R., BG, V. K., Carneiro, G., & Reid, I. (2016). Unsupervised cnn for single view depth estimation: Geometry to the rescue. In *Proceedings of the European Conference on Computer Vision (ECCV)* (pp. 740–756).

Geiger, A., Lenz, P., & Urtasun, R. (2012). Are we ready for autonomous driving? The kitti vision benchmark suite. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 3354–3361). IEEE.

Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F. A., & Brendel, W. (2019). Imagenet-trained cnns are biased towards texture; Increasing shape bias improves accuracy and robustness. In *Proceedings of the International Conference on Learning Representations (ICLR)* (pp. 248–255).

Godard, C., Aodha, O. M., Firman, M., & Brostow, G. J. (2019). Digging into self-supervised monocular depth estimation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (pp. 3828–3838).

Godard, C., Mac Aodha, O., & Brostow, G. J. (2017). Unsupervised monocular depth estimation with left-right consistency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 270–279).

Guizilini, V., Hou, R., Li, J., Ambrus, R., & Gaidon, A. (2020). Semantically-guided representation learning for self-supervised monocular depth. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Gurram, A., Tuna, A., Shen, F., Urfalioglu, O., & López, A. (2021). *Monocular depth estimation through virtual-world supervision and real-world sfm self-supervision*. arxiv 2021. arXiv preprint arXiv:2103.12209

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 770–778).

He, L., Wang, G., & Hu, Z. (2018). Learning depth from single images with deep neural network embedding focal length. *IEEE Transactions on Image Processing, 27*(9), 4676–4689.

Hirschmuller, H. (2007). Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 30*(2), 328–341.

Hoffman, J., Tzeng, E., Park, T., Zhu, J. Y., Isola, P., Saenko, K., Efros, A. A., & Darrell, T. (2018). Cycada: Cycle consistent adversarial domain adaptation. In *Proceedings of the International Conference on Machine Learning (ICML)* (pp. 1989–1998).

Hu, J., Zhang, Y., & Okatani, T. (2019). Visualization of convolutional neural networks for monocular depth estimation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (pp. 3869–3878).

Huang, Y. K., Wu, T. H., Liu, Y. C., & Hsu, W. H. (2019). Indoor depth completion with boundary consistency and self-attention. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV) Workshops* (pp. 1070–1078).

Jiao, J., Cao, Y., Song, Y., & Lau, R. (2018). Look deeper into depth: Monocular depth estimation with semantic booster and attention-driven loss. In *Proceedings of the European Conference on Computer Vision (ECCV)* (pp. 53–69).

Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Kirillov, A., Girshick, R., He, K., & Dollar, P. (2019a). Panoptic feature pyramid networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 6399–6408).

Kirillov, A., He, K., Girshick, R., Rother, C., & Dollár, P. (2019b). Panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 9404–9413).

Klingner, M., Termöhlen, J. A., Mikolajczyk, J., & Fingscheidt, T. (2020). Self-supervised monocular depth estimation: Solving the dynamic object problem by semantic guidance. In *Proceedings of the European Conference on Computer Vision (ECCV)* (pp. 582–600). Springer.

Kundu, J. N., Lakkakula, N., & Babu, R. V. (2019) Um-adapt: Unsupervised multi-task adaptation using adversarial cross-task distillation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (pp. 1436–1445).

Laina, I., Rupprecht, C., Belagiannis, V., Tombari, F., & Navab, N. (2016). Deeper depth prediction with fully convolutional residual networks. In *2016 Fourth international conference on 3D vision (3DV)* (pp. 239–248). IEEE.

Lambert, J., Zhuang, L., Sener, O., Hays, J., & Koltun, V. (2020). MSeg: A composite dataset for multi-domain semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 2879–2888).

Li, R., Mao, Q., Wang, P., He, X., Zhu, Y., Sun, J., & Zhang, Y. (2020). Semantic-guided representation enhancement for self-supervised monocular trained depth estimation. arXiv preprint arXiv:2012.08048

Li, W., Saeedi, S., McCormac, J., Clark, R., Tzoumanikas, D., Ye, Q., Huang, Y., Tang, R., & Leutenegger, S. (2018). Interiornet: Mega-scale multi-sensor photo-realistic indoor scenes dataset. In *Proceedings of the British Machine Vision Conference (BMVC)*.

Li, Z., & Snavely, N. (2018). Megadepth: Learning single-view depth prediction from internet photos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 2041–2050).

Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *Proceedings of the European Conference on Computer Vision (ECCV)* (pp. 740–755).

Liu, K. C., Shen, Y. T., Klopp, J. P., & Chen, L. G. (2019). What synthesis is missing: Depth adaptation integrated with weak supervision for indoor scene parsing. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (pp. 7345–7354).

Lopez-Rodriguez, A., Busam, B., & Mikolajczyk, K. (2020). Project to adapt: Domain adaptation for depth completion from noisy and sparse sensor data. In *Proceedings of the Asian Conference on Computer Vision*.

Lopez-Rodriguez, A., & Mikolajczyk, K. (2020). Desc: Domain adaptation for depth estimation via semantic consistency. In *British Machine Vision Conference (BMVC)*.

Luo, Y., Ren, J., Lin, M., Pang, J., Sun, W., Li, H., & Lin, L. (2018). Single view stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 155–163).

Mao, X., Li, Q., Xie, H., Lau, R. Y., Wang, Z., & Paul Smolley, S. (2017). Least squares generative adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (pp. 2794–2802).

Mayer, N., Ilg, E., Hausser, P., Fischer, P., Cremers, D., Dosovitskiy, A., & Brox, T. (2016). A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 4040–4048).

Meng, Y., Lu, Y., Raj, A., Sunarjo, S., Guo, R., Javidi, T., Bansal, G., & Bharadia, D. (2019). Signet: Semantic instance aided unsupervised 3d geometry perception. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 9810–9820).

Menze, M., & Geiger, A. (2015). Object scene flow for autonomous vehicles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 3061–3070).

Mohan, R., & Valada, A. (2021). Efficientps: Efficient panoptic segmentation. *International Journal of Computer Vision, 129*(5), 1551–1579.

Mousavian, A., Pirsiavash, H., & Košecká, J. (2016). Joint semantic segmentation and depth estimation with deep convolutional networks. In *2016 Fourth International Conference on 3D Vision (3DV)* (pp. 611–619). IEEE.

Nath Kundu, J., Krishna Uppala, P., Pahuja, A., & Venkatesh Babu, R. (2018). Adadepth: Unsupervised content congruent adaptation for depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 2656–2665).

Pilzer, A., Lathuiliere, S., Sebe, N., & Ricci, E. (2019a). Refine and distill: Exploiting cycle-inconsistency and knowledge distillation for unsupervised monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 9768–9777).

Pilzer, A., Lathuilière, S., Xu, D., Puscas, M. M., Ricci, E., & Sebe, N. (2019b). Progressive fusion for unsupervised binocular depth estimation using cycled networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 42,* 2380–2395.

PNVR, K., Zhou, H., & Jacobs, D. (2020). Sharingan: Combining synthetic and real data for unsupervised geometry estimation. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)* (pp. 234–241).

Ros, G., Sellart, L., Materzynska, J., Vazquez, D., & Lopez, A. M. (2016). The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 3234–3243).

Roy, S., Siarohin, A., Sangineto, E., Bulo, S. R., Sebe, N., & Ricci, E. (2019). Unsupervised domain adaptation using feature-whitening and consensus loss. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 9471–9480).

Saeedan, F., & Roth, S. (2021). Boosting monocular depth with panoptic segmentation maps. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision* (pp. 3853–3862).

Saha, S., Obukhov, A., Paudel, D. P., Kanakis, M., Chen, Y., Georgoulis, S., & Van Gool, L. (2021). Learning to relate depth and semantics for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 8197–8207).

Saito, K., Ushiku, Y., & Harada, T. (2017). Asymmetric tri-training for unsupervised domain adaptation. In *Proceedings of the International Conference on Machine Learning (ICML)* (pp. 2988–2997).

Sajjadi, M., Javanmardi, M., & Tasdizen, T. (2016). Regularization with stochastic transformations and perturbations for deep semi-supervised learning. In *Advances in Neural Information Processing Systems (NeurIPS)* (pp. 1163–1171).

Saxena, A., Sun, M., & Ng, A. Y. (2008). Make3d: Learning 3d scene structure from a single still image. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 31*(5), 824–840.

Silberman, N., Hoiem, D., Kohli, P., & Fergus, R. (2012). Indoor segmentation and support inference from rgbd images. In *Proceedings of the European Conference on Computer Vision (ECCV)* (pp. 746–760).

Tsai, Y. H., Hung, W. C., Schulter, S., Sohn, K., Yang, M. H., & Chandraker, M. (2018). Learning to adapt structured output space for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 7472–7481).

Tzeng, E., Hoffman, J., Saenko, K., & Darrell, T. (2017). Adversarial discriminative domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 7167–7176).

Vu, T. H., Jain, H., Bucher, M., Cord, M., & Pérez, P. (2019). Dada: Depth-aware domain adaptation in semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (pp. 7364–7373).

Wu, Y., Kirillov, A., Massa, F., Lo, W. Y., & Girshick, R. (2019). *Detectron2*. https://github.com/facebookresearch/detectron2

Xie, J., Girshick, R., & Farhadi, A. (2016). Deep3d: Fully automatic 2d-to-3d video conversion with deep convolutional neural networks. In *Proceedings of the European Conference on Computer Vision (ECCV)* (pp. 842–857).

Yin, Z., & Shi, J. (2018). Geonet: Unsupervised learning of dense depth, optical flow and camera pose. In *Proceedings of the IEEE/CVF*

Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 1983–1992).

You, K., Wang, X., Long, M., & Jordan, M. (2019). Towards accurate model selection in deep unsupervised domain adaptation. In *International Conference on Machine Learning* (pp. 7124–7133). PMLR.

Zhang, Z., Cui, Z., Xu, C., Jie, Z., Li, X., & Yang, J. (2018). Joint task-recursive learning for semantic segmentation and depth estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)* (pp. 235–251).

Zhao, S., Fu, H., Gong, M., & Tao, D. (2019). Geometry-aware symmetric domain adaptation for monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 9788–9798).

Zhao, Y., Kong, S., Shin, D., & Fowlkes, C. (2020). Domain decluttering: Simplifying images to mitigate synthetic-real domain shift and improve depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 3330–3340).

Zheng, C., Cham, T. J., & Cai, J. (2018). T2net: Synthetic-to-realistic translation for solving single-image depth estimation tasks. In *Proceedings of the European Conference on Computer Vision (ECCV)* (pp. 767–783).

Zhou, B., Kalra, N., & Krähenbühl, P. (2020). Domain adaptation through task distillation. In *Proceedings of the European Conference on Computer Vision (ECCV)* (pp. 664–680). Springer.

Zhou, T., Brown, M., Snavely, N., & Lowe, D. G. (2017). Unsupervised learning of depth and ego-motion from video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 1851–1858).

Zhu, S., Brazil, G., & Liu, X. (2020). The edge of depth: Explicit constraints between segmentation and depth. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 13116–13125).