Adversarial Learning for Depth and Viewpoint Estimation from a Single Image

Saddam Abdulwahab, Hatem A. Rashwan, Miguel Angel Garcia, Mohammed Jabreel, Sylvie Chambon and Domenec Puig

***Abstract*—Estimating a depth map and, at the same time, predicting the 3D pose of an object from a single 2D color image is a very challenging task. Depth estimation is typically performed through stereo vision by following several time-consuming stages, such as epipolar geometry, rectification and matching. Alternatively, when stereo vision is not useful or applicable, depth relations can be inferred from a single image as studied in this paper. More precisely, deep learning is applied in order to solve the problem of estimating a depth map from a single image. Then, that map is used for predicting the 3D pose of the main object depicted in the image. The proposed model consists of two successive neural networks. The first network is based on a Generative Adversarial Neural network (GAN). It estimates a dense depth map from the given color image. A Convolutional Neural Network (CNN) is then used to predict the 3D pose from the generated depth map through regression. The main difficulty to jointly estimate depth maps and 3D poses using deep networks is the lack of training data with both depth and viewpoint annotations. This contribution assumes a cross-domain training procedure with 3D CAD models corresponding to objects appearing in real images in order to render depth images from different viewpoints. These rendered images are then used to guide the GAN network to learn the mapping from the image domain to the depth domain. By exploiting the dataset as a source of training data, the proposed model outperforms state-of-the-art models on the PASCAL 3D+ dataset. The code of the proposed model is publicly available at https://github.com/SaddamAbdulrhman/Depth-and-Viewpoint-Estimation/tree/master.**

Deep learning depth prediction pose estimation generative adversarial networks.

I. Introduction

Identifying objects and, more generally, understanding the scene of an input image is a challenging goal in computer vision. It is useful for many applications, such as face recognition, video surveillance and robotics. Inferring 3D shapes and pose from a single perspective is a fundamental capability of human vision, although a tough task for computer vision. The appearance of an object in an image dramatically depends on its intrinsic characteristics (e.g., texture and color/albedo), and extrinsic characteristics related to the acquisition (e.g., camera pose and gamma correction conditions). The appearance of objects significantly changes with their pose [1]. Estimating a depth map from a 2D image is an important step in order to determine the 3D pose of the objects present in a scene.

Saddam Abdulwahab, Hatem A. Rashwan, Mohammed Jabreel and Domenec Puig are with Dept. of Computer Engineering and Mathematics, Universitat Rovira i Virgil, Tarragona, Spain; Miguel Angel Garcia is with Dept. of Electronic and Communications Technology, Universidad Autnoma de Madrid, Spain; and Sylvie Chambon is with IRIT-CNRS, ENSEEIHT, University of Toulouse, Toulouse, France .

In general, estimating a 3D pose requires the solution of two problems: (1) *generating the best depth image from a single image,* (2) *estimating the correct pose of the main object in the 3D scene.*

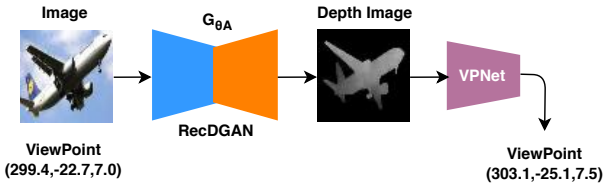

Fig. 1: Proposed framework for simultaneous depth and 3D viewpoint estimation (Azimuth, Elevation,Distance), in the test stage.

Recently, with the outstanding progress of deep learning, several methods based on deep networks have been proposed for 3D shape generation from a single color image of an object [2], [3]. These methods use different deep models for image-to-image translation to learn the mappings among multiple domains, such as Fully Convolutional Networks (FCN) [4], U-Net networks [5], and Generative Adversarial Networks (GAN) [6], [7]. Furthermore, Convoultional Neural Networks are also used for estimating 3D poses [8], [9]. The majority of deep network models for depth and viewpoint estimation are trained with input color images and depth images captured with depth cameras or LiDAR sensors [10], [11]. However, LiDARs are very costly, and most depth cameras have serious limitations in real environments, such as the synchronization of the optical and imaging elements [12].

In this work, we propose to use a GAN network, a cutting-edge technique for image-to-image translation, as the baseline network for predicting a depth image from a single color image. However, with the lack of annotated training data for both depth images of objects and 3D poses, we propose a cross-domain training model [13]. In particular, we use 3D CAD models for rendering depth images from different viewpoints. The obtained depth images and their pose information are used to train the proposed network. Consequently, the proposed model consists of two successive networks. The first network (RecDGAN) estimates a depth image from the input image. This network embodies two generators and one discriminator. The first generator learns to map the RGB image domain into the depth domain. In order to enforce that the generated depth image be an image-based representation of the input RGB image, the second generator reconstructs the original

RGB image from the generated depth image by using a reconstruction loss function [14]. To make the generated depth image closer to the depth domain, a discriminator is trained with a GAN loss. In turn, the second network (VPNet) is a regression CNN network that predicts the 3D pose of the main object depicted in the input image (i.e., elevation and azimuth angles along with the distance from the camera to the object). The two networks are integrated into a single pipeline to solve the two problems of depth and pose estimation simultaneously. Fig.1 shows the proposed framework for simultaneous depth and 3D pose estimation.

To the best of our knowledge, this work is the first attempt to use a cross-domain training deep network model for estimating both the depth and 3D pose of the main object depicted in a 2D image. The main contributions of this paper are the following:

- The design of a GAN network with a loss function for feature matching that allows the system to generate a dense depth image from a single 2D color image of an object.
- A novel regression network to predict the 3D pose from the generated depth image.
- The integration of the two networks into a single pipeline to solve the problems of generating a depth image and estimating the 3D pose from a single color image.

This paper is organized as follows. Section 2 describes the related work in this field. Section 3 describes the proposed methodology to estimate a depth image and its 3D pose using both RecDGAN and VPNet. Section 4 describes experimental results and the obtained performance. Finally, Section 5 concludes this work, suggesting future lines of research.

II. Related work

In this section, we present a quick review of previous works related to depth estimation and 3D pose estimation from a single image by using classical computer vision techniques and deep learning techniques. To our knowledge, no methods have been proposed so far to estimate both features simultaneously like our model. Thus, the previous work is divided into two parts related to these sub-problems: 1) depth estimation, 2) viewpoint estimation.

A. Depth Estimation

This subsection focuses on accurate methods to solve the depth estimation problem. The start with, the work presented in [15] proposes a fully automatic 2D-to-3D integrated face reconstruction approach to reconstruct a personalized 3D face model from a single frontal face image with a neutral expression and normal gamma correction. The reconstructed 3D faces are then used for face recognition. However, this method cannot effectively improve the recognition performance of near-profile views due to the unreliable synthesis of the profile virtual views. This indicates that the facial features on the frontal views are not associated with the height information of face shapes [16].

In [17], the proposed model takes a pixel from an original image as a sample point and estimates the depth of the other pixels. This model is not able to accurately extract global structure from a single image due to the limitations of only processing local information. Saxena et al. [18] developed a discriminatively trained Markov Random Field (MRF) model for depth estimation from single monocular images. This model uses monocular cues at multiple spatial scales and also incorporates interaction terms that model relative depths at different scales. In addition to a Gaussian MRF model, they also presented a Laplacian MRF model in which Maximum a Posteriori (MAP) inference can be done efficiently using linear programming. However, the system relies on horizontal alignment of images and suffers in less controlled settings. In [19], Make3D is proposed to generate a 3D model from a single image. However, the system has a poor performance in uncontrolled settings due to its dependence on the horizontal alignment of images. Saxena et al. [20] utilize color, texture and other visual cues at multiple scales to build the relationship between image patches and adjacent depth map spots using a Markov stochastic model in order to calculate the depth map corresponding to the original image.

Nowadays, with the significant progress of deep learning models, several approaches based on deep networks have been proposed to predict depth maps from a single image. In particular, [21] presents a framework for depth and surface normal estimation from single monocular images. It consists of a regression stage using a deep CNN model to learn the mapping from multi-scale image patches to depth or surface normal values at the super-pixel level (the SLIC algorithm [22] is used to obtain the super-pixels). They then refine the estimated super-pixel depth or surface normal to the pixel level by exploiting the potentials on the depth or surface normal map, which include a data term, a smoothness term among super-pixels and an auto-regression term characterizing the local structure of the estimation map. In [10], an approach is presented for estimating depth from a single image by combining information from both global and local views. They use two deep networks: one that estimates the global depth structure and predicts the depth of the scene at a global level, and another that takes the first network output as additional first-layer image features in order to edit the global prediction as to incorporate finer-scale details. Moreover, they apply a scale-invariant error to measure depth relations rather than scale. Furthermore, the network is trained using a loss function that explicitly accounts for depth relations between pixel locations, in addition to the point-wise error. However, the system suffers from a low performance in estimating the surface depths. Furthermore, in [23], a three-layer CNN trained with a per-pixel Euclidean loss is presented to transform the given color image to a geometrically meaningful output image. Besides, this method uses Conditional Random Fields (CRF) as a loss layer to enforce local consistency in the output image.

Finally, a depth generative adversarial network (DepthGAN) has been proposed in [24] by using the advantage of a Fully Convolutional Residual Network (FCRN) and combining it with a GAN network. The authors also present a new loss function that includes a scale-invariant (SI) error for solving the scale invariance problem that arises when predicting depth from a single image. Moreover, they use a structural similarity (SSIM) loss function to derive both the relative and the

absolute distances of objects based on the textural structure in the scene.

B. Viewpoint Estimation

This subsection overviews the most effective methods to solve the viewpoint estimation problem. In [25], average shading gradients (ASG) are proposed. The gradient normals of all lighting directions are averaged to cope with the unknown lighting of the query image. The main advantage of ASG is to ignore color and texture in the expression of the 3D model shape. Image gradients are then matched with ASG images to estimate a 3D pose. Unfortunately, image gradients are still affected by image textures and background. Following a different approach, in [26]–[28], a collection of depth images of 3D models rendered from different viewpoints are used to detect curvilinear features. The authors in [27], [28] propose three main steps. First, the ridges and valleys of the depth images rendered from the 3D model are detected. In order to cope with the texture and background in the 2D images, curvilinear features are extracted with a multiscale scheme. These features are then refined by only keeping in-focus features. The final step determines the correct 3D pose using a repeatable K-NN [27] and SVM [28] in the registration algorithm (i.e., instance-based learning) until finding the closest view. In [29] the authors propose a network for depth prediction that uses a sequence of three scales to generate features and capture image details. They make a consistent global prediction and then utilize it with iterative local refinements. In that way, the local networks are aware of their location within the global scene and use this information in their refined predictions. Moreover, they upsample the refined predictions to a higher resolution.

Our model for viewpoint estimation is inspired by recent works [30], [31] to learn how to predict the viewpoints based on a CNN. In [30], the authors introduced a deep model based on CNN for monocular viewpoint estimation by using keypoint information provided by humans at inference time to accurately estimate the viewpoint of an object. Their work aims to capture the relation between viewpoints of objects and key-points for specific objects. They exploited this relationship and refined an existing coarse pose estimation using keypoint predictions. But post-refinement processes are still required to compensate for the accuracy sacrificed by the discretization. In [30], the problem of pose estimation was designed as a classification method. Alternatively, the problem has recently been modeled and solved by regression deep neural networks. In [31], the authors proposed a CNN-based approach for monocular viewpoint estimation based on the structure of the viewpoint space when designing regression losses and non-linear activation functions. This approach is more advantageous to handle the challenging case of nearly symmetric objects. Also, they used a data augmentation strategy designed to capture perturbations in the viewpoint space.

Other researchers have also successfully used deep learning to solve the pose estimation problem. However, with the lack of data during training, the solution is taking advantage of CAD models and additional annotations to generate more training data. For instance, [32] rendered millions of synthetic images from 3D models and then used them to train a CNN model for viewpoint estimation of real images. However, our model does not require the use of all these data sources. To mitigate the low amount of data, we render depth images from 3D models based on the viewpoints of real images. We then apply data augmentation techniques to all generated images to increase the number of training samples under different conditions. This generates new realistic data from the real images and 3D models.

In [33], a deep CNN performs full 3D hand pose estimation from single color images. This approach consists of three deep networks. The first network applies segmentation to localize the hand in the image. Based on that, the second network localizes hand key points in the 2D images. The third network finally derives the 3D hand pose from the 2D key points. Although this approach uses a large synthetic dataset, its performance seems mostly limited by the lack of an annotated large-scale dataset with real-world images and different pose statistics.

Recently, the authors of [34] proposed a method for estimating the pose of an object from a single image using multiple-viewpoint correspondence based on CNN networks. Initially, they find a consistent local feature description of the object's parts in the input RGB image. After that, they use these descriptors along with the key points obtained from the renders of a fixed 3D template model to create basic depth maps of a particular monochrome real image. Finally, a pose estimation network predicts the 3D pose of the object using these correspondence maps. In [35] the authors proposed a method for estimating of 3D structures and camera projection using symmetry and/or Manhattan structure cues from a single image or multiple images in the same category. They recover the camera projection from a single image using the Manhattan structure. They also use multiple images to exploit symmetry without requiring the Manhattan structure for 3D reconstruction, since the Manhattan structure can be hard to observe from a single image due to occlusion.

In [36], a method for 3D object detection and pose estimation from a single image was proposed using a deep CNN and Geometry. To estimate the full 3D pose and dimensions of an object surrounded by a 2D bounding box, they used a discrete-continuous CNN architecture with a loss function for orientation prediction and a practical choice of box dimensions as regression parameters. The method estimates the 3D bounding boxes without additional 3D shape models or sampling strategies with complex pre-processing pipelines. Although this method properly estimates object orientation and localizes the objects in 3D from an image, it depends on different geometric constraints, such as shape priors or occlusion patterns to infer 3D bounding boxes. In turn, in [37], another method was introduced for retrieving 3D models of objects in the wild. This approach consists of two networks. The first network estimates the 3D pose of an object and the second network uses synthetic depth images rendered from 3D models based on the 3D pose estimated from the first network in order to retrieve 3D models that accurately represent the geometry of objects present in RGB images. This is done

by comparing the learned image descriptors of RGB images against those of the rendered depth images using a CNN-based multi-view metric learning approach.

In this section, we summarize the state of the art for depth and 3D pose estimation from a single image through classical computer vision techniques and deep learning techniques.

The approaches based on deep learning yield the most accurate results. Thus, we propose a method based on a deep model, RecDGAN, to obtain both the depth and 3D pose from a single image.

III. Proposed Methodology

The proposed model is based on two different generators coupled together. Each generator is able to map from a domain to another. In particular, the first generator learns to map from an RGB image to a depth image. The latter is forced to be an image-based representation of the input RGB image by reconstructing the same RGB image through the second generator, with a feature matching loss being used as a reconstruction loss function. During training, the depth image estimated by the first generator is compared to a depth image generated after rendering a synthetic 3D model through a discriminator network. With the two losses, each generator in the proposed model is able to learn the mapping from its input domain to the output domain and to discover relations between them. The generated depth image is also fed into a regression CNN network that estimates the 3D pose of the main object depicted in the depth image.

In this paper, we propose to consider depth image estimation as an image-to-image translation task as proposed in [14], [38]. In [38], there are two generators and one discriminator, whereas in [14], there are two generators and two discriminators. In our model, we apply two generators and one discriminator in addition to a regression CNN network that predicts the 3D pose of the main object depicted in the input image (i.e., elevation and azimuth angles along with the distance from the camera to the object). The viewpoint estimation network will help the generator to find the correct orientation of the object. In addition, we use a multi-scale feature matching loss function based on CNN to improve the performance of the generators. It makes the generated depth image closer to the depth map domain and the reconstructed image closer to the real image.

This section describes the proposed system and its training procedure. Fig.2 shows the architecture of the proposed system. It is constituted by two main sub-models: a depth generator based on a Generative Adversarial Network (GAN), and a viewpoint estimator from the generated depth image based on a CNN.

We formulate the problem in subsection A. The remaining subsections explain each part of the proposed model in detail.

A. Problem Formulation

Let $A \in \mathbb{A}$ be a 2D color image. The problem of generating its corresponding depth image, $B \in \mathbb{B}$, can be formally defined as a function $f : \mathbb{A} \rightarrow \mathbb{B}$ that maps elements from domain $\mathbb{A}$ to elements in its co-domain $\mathbb{B}$. Similarly, we can formally define the problem of estimating the viewpoint of a 2D image as a function $g : \mathbb{A} \rightarrow \mathbb{R}^3$ that takes as input a 2D color image and predicts three viewpoint values, namely: azimuth, elevation and distance. We introduce a multi-task deep learning-based system to solve the two aforementioned sub-problems. Specifically, The proposed system consists of two generators, $G_{\theta_A}(A)$ and $G_{\theta_B}(\hat{B})$, a discriminator $D_{\theta_D}(\alpha)$, and a viewpoint estimator $V_{\theta_V}(\hat{B})$, where $\hat{B}$ is the depth image generated by G_{θ_A} and $\alpha \in \{B \times \hat{B}\}$. A feature matcher $fmrecogan(A, \hat{A})$ is used to comapre the image reconstructed by G_{θ_B}, $\hat{A} = G_{\theta_B}(\hat{B})$, with the input color image A. The next subsections explain in detail the architecture of our system, its sub-models, and the training procedure.

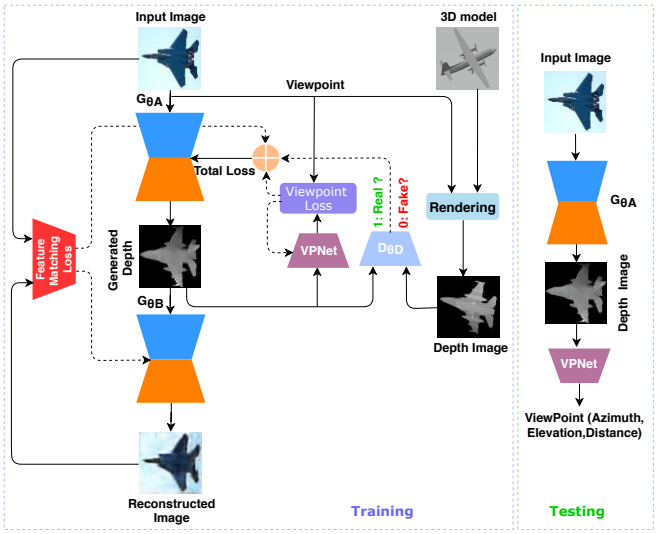

Fig. 2: Proposed RecDGAN system to generate depth images and 3D poses from 2D color images. Details about G_{θ_A}, G_{θ_B} and D_{θ_D} are given in section 3.2. $VPNet$ and Feature Matching Loss are detailed in sections 3.3 and 3.4, respectively.

B. Generative adversarial networks (GANs)

The generative adversarial network framework is a supervised deep learning model proposed by Goodfellow et al. [39], originally focusing on image generation and manipulation tasks for training an image synthesis model aiming at the generation of artistic images. It is implemented by two neural networks: a generator and a discriminator. Many variants based on GANs have already been developed [40], [41]. They have been applied to practical image generation problems [42]–[44]. Recently, conditional GANs, which are an extension of GANs, have shown great success in using conditional adversarial networks to learn the loss function for image-to-image translation tasks [6], [38]. All these methods have successfully led to the estimation of transformation networks from one image domain to another.

In the present work, generator G_{θ_A} takes a real input color image and maps it to a depth image, while generator G_{θ_B} takes the depth image generated by G_{θ_A} and maps it to a color image. The input of the discriminator D_{θ_D} is a depth image rendered from a training dataset and the depth image predicted

by G_{θ_A}. D_{θ_D} estimates the probability that both depth images are similar. The discriminator network of the GAN assesses whether the predicted depth image is likely to belong to the depth image domain or not.

1) Generative Networks: This subsection describes the generative neural networks G_{θ_A} and G_{θ_B}. Both have an identical structure. G_{θ_A} learns the mapping from an input color image to its corresponding depth image. The input of G_{θ_A} is a 2D color image, A, and it generates a depth image, $\hat{B}$, which is then fed to G_{θ_B} to estimate a 2D color image $\hat{A} = G_{\theta_B}(G_{\theta_A}(A))$, where $\hat{A}$ is a reconstruction of the original 2D image A. For comparing the two images, we use a loss function based on feature matching , which is explained in detail in Section 3.4. The objective loss function of the generator is:

$$\mathfrak{L}_{rcon}(\theta_A, \theta_B, \mathbb{A}) = \mathbb{E}_{A\in\mathbb{A}, \hat{A}=G_{\theta_B}(G_{\theta_A}(A))}[\Delta(A, \hat{A})], \quad (1)$$

where θ_A and θ_B are the parameters of the two generators and Δ is a measure of discrepancy between the two images.

The architecture of our generative network is shown in Fig.3. It consists of an encoder and a decoder. Inspired in [14], the encoder of each generator is composed of five convolution layers with 4×4 filters, stride 2 and padding 1. Each convolution layer is followed by batch normalization (BN) except for C_{n1}, and by *LeakyReLU* [23], [45]. In turn, the decoder part is composed of five deconvolution layers with a filter size of 4×4, stride 2 and padding 1. Each layer is followed by *ReLU* and BN except for D_{n5}, which applies a sigmoid. The output is a depth image of size $64 \times 64 \times 1$. An example of the features extracted and generated by the generator layers is shown in Fig.4.

2) Discriminator Network: The generator G_{θ_A} aims at yielding depth images that belong to domain $\mathbb{B}$. To model this additional constraint, we train a discriminator to determine whether the depth images estimated by the generator G_{θ_A} are real depth images or not.

The architecture of the discriminator is shown in Fig.5. It consists of an encoder identical to the one of the generator, followed by an output logistic unit.

$$\mathfrak{L}_{dis}(\theta_D|\theta_A, \mathbb{B}, \mathbb{A}) = -\mathbb{E}_B[log(p_D(B))], \quad (2)$$

where p_D represents the prediction entropy of the discriminator with the real depth B belonging to the domain $\mathbb{B}$, i.e. $B \in \mathbb{B}$. θ_A and θ_D are the parameters of the first generator and discriminator, respectively.

The prediction cross-entropy of the discriminator with the estimated depth image, $\hat{B} = G_{\theta_A}(A)$, can be defined as:

$$\mathfrak{L}_{\mathfrak{adv}}(\theta_A, \mathbb{A}|\theta_D) = -\mathbb{E}_{A\in\mathbb{A}}[log(1 - p_D(G_{\theta_A}(A))]. \quad (3)$$

The optimizer will fit D to maximize the loss values for real depth images rendered from 3D CAD models (by minimizing $\log(p_D(B))$) and to minimize the loss values for estimated depth images (by minimizing $\log(1-p_D(G_{\theta_A}(A)))$. The generator and discriminator networks are optimized concurrently, one optimization step for both networks at each iteration, where G_{θ_A} tries to generate an accurate depth estimation and D learns how to discriminate between the synthetic and the real depth maps.

Thus, the adversarial loss used for training the model is:

$$\begin{aligned}\mathfrak{L}_{\mathfrak{gan}}(\theta_A, \theta_D, \mathbb{A}, \mathbb{B}) = \mathfrak{L}_{dis}(\theta_D|\theta_A, \mathbb{B}, \mathbb{A}) \\ +\mathfrak{L}_{\mathfrak{adv}}(\theta_A, \mathbb{A}|\theta_D).\end{aligned} \quad (4)$$

GAN can often be defined as a minimax game in which the generator wants to minimize $\mathfrak{L}_{\mathfrak{gan}}$ while the discriminator wants to maximize it.

C. Viewpoint Estimation Network

The second goal of our system is to use the generated depth image of an object to estimate its correct viewpoint. The motivation for estimating the 3D pose of a single depth image is that depth measurement avoids the ambiguity caused by perspective projection in 2D images. In addition, depth images are invariant to lighting conditions. To do so, we train a regression neural network, VPNet, to estimate the viewpoint from the depth image generated by G_{θ_A}. The architecture of VPNet is shown in Fig.6. Again, it consists of an encoder identical to the ones of the discriminator and the generator followed by a linear layer of three units. VPNet is trained to minimize the following loss function:

$$\begin{aligned}\mathfrak{L}_{\mathfrak{vp}}(\theta_V, V, \mathbb{A}|\theta_A) = \\ \mathbb{E}_{(v,A)\in(V,\mathbb{A})}[\Delta_v(v, \hat{v} = VPNet(G_{\theta_A}(A)))],\end{aligned} \quad (5)$$

where v is the real 3D pose, $\hat{v}$ is the estimated one, Δ_V is a measure of the difference between the real value and the estimated value, and θ_V is the set of parameters of the viewpoint estimator. We use the mean square error as the difference measure Δ_v between two 3D poses. For more details, see the supplementary materials.

D. Loss Function for Feature Matching

The proposed loss function for feature matching depends on the features extracted from both the input real image A and the reconstructed image $\hat{A}$ from G_{θ_B}, by taking into account color and texture. The usual comparison functions, such as the L_1 or L_2 norms, as proposed in the cycleGAN network [14], are not effective in order to measure similarity between two images. In addition, in a normal GAN, the discriminator and generator are always in a tug of war to undercut each other. Mode collapse and gradient diminishing are often explained as an imbalance between the discriminator and the generator.

Thus, adding a new discriminator will increase the model complexity and may also overfit the generator network. Therefore, we use feature matching based on CNN inspired in [14] by replacing the L1-norm by a feature matching network in order to achieve a more accurate comparison of the input and reconstructed images. In particular, we compare the multi-scale features extracted from different CNN layers of the input RGB image with the corresponding ones extracted from the RGB image generated by G_{θ_B}, and then the network attempts to minimize the difference between the corresponding features. Indeed, by replacing the L1-norm with a feature matching loss causes training to be more stable and converge faster.

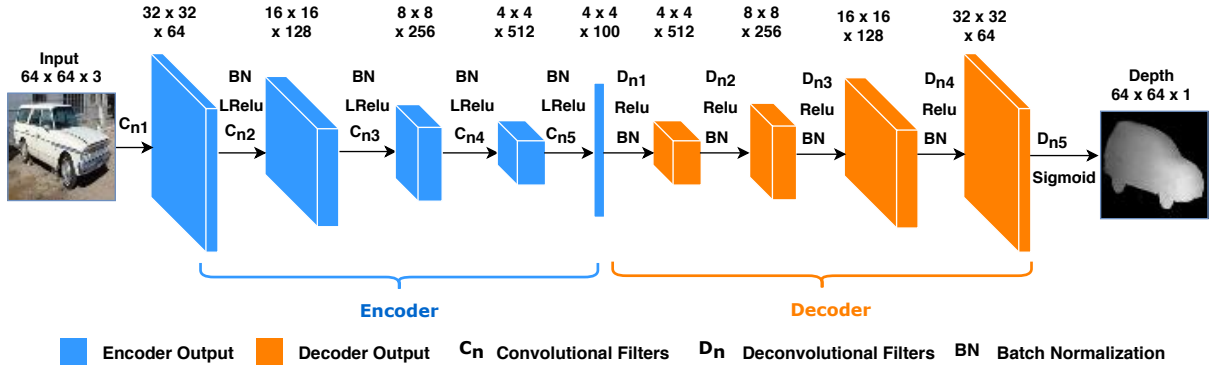

Fig. 3: Architecture of the generator network.

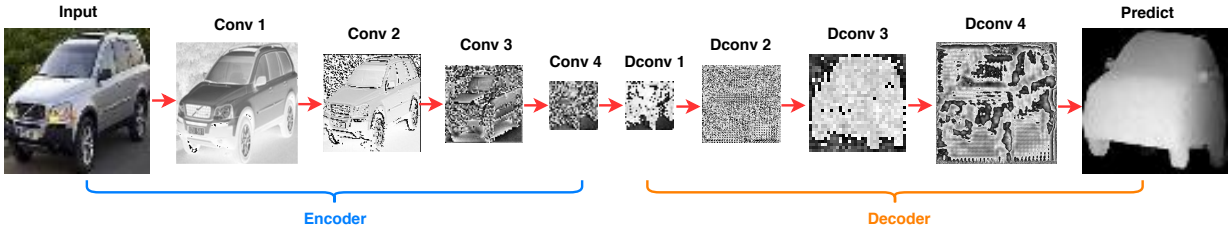

Fig. 4: Features extracted by each layer of the generator network.

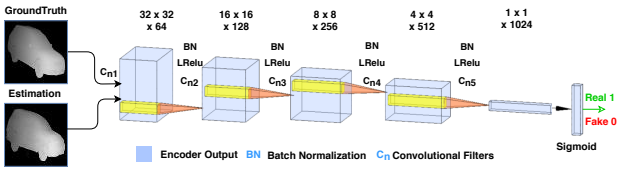

Fig. 5: Architecture of the discriminator network.

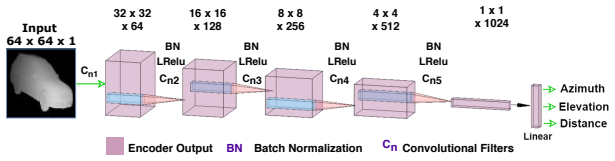

Fig. 6: Architecture of the viewpoint estimator network.

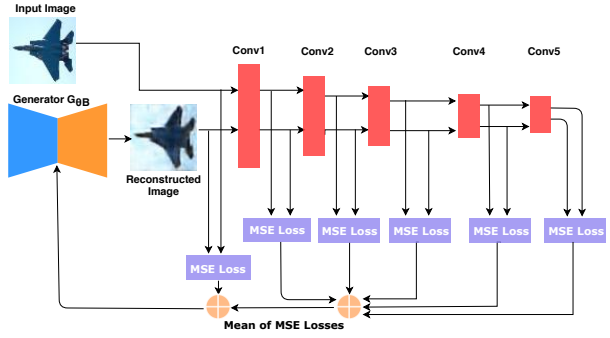

Fig. 7: Feature matching loss architecture.

We use a CNN of five layers. To calculate the similarity between the two input images, the feature matching loss ($\mathfrak{L}_{\texttt{recon}}$) is based on the features extracted per layer from the input real image A and the ones from the reconstructed image $\hat{A}$. The aggregated loss function $\mathfrak{L}_{\texttt{recon}}$ is computed between A and $\hat{A}$ as:

$$\mathfrak{L}_{\texttt{recon}}(A, \hat{A}|\theta_B) = \frac{1}{N}\sum_{i=1}^{N} f(A_{L_{si}(i)} - \hat{A}_{L_{sr}(i)}), \quad (6)$$

where N is the number of layers (N is empirically set to 5 in this work), f is the MSE error, $A_{L_{si}(i)}$ is a loss layer of the features from the real image and $\hat{A}_{L_{sr}(i)}$ is a loss layer of the features from the estimated image.

E. Final Objective Function

The final objective function, i.e. the training loss, of our learning algorithm is defined as:

$$\begin{aligned}\mathfrak{L}(\theta_A, \theta_B, \theta_D, \theta_V, \mathbb{A}, \mathbb{B}, \hat{A}, V) = \\ \lambda_{gan}[\mathfrak{L}_{\texttt{gan}}(\theta_A, \theta_D, \mathbb{A}, \mathbb{B})]+ \\ \lambda_{vp}[\mathfrak{L}_{\texttt{vp}}(\theta_V, V, \mathbb{A}|\theta_A)]+ \\ \lambda_{recon}[\mathfrak{L}_{\texttt{recon}}(\mathbb{A}, \hat{A}|\theta_B)],\end{aligned} \quad (7)$$

where λ_{gan}, λ_{vp} and λ_{recon} are hyper-parameters weighing the importance of the discriminator loss, adversarial loss, viewpoint loss and the loss function for feature matching. In our model, λ_{gan}=λ_{vp}=λ_{recon} = 1 yields the best accuracy for 3D pose estimation.

IV. Experiments and Results

This section describes the experiments performed to evaluate the proposed model, in addition to the dataset, the data augmentation and the evaluation measures used in the experiments.

A. Dataset

In this work, a comprehensive set of experiments have been conducted in order to validate the performance of the proposed model on the public PASCAL3D+ dataset [46], which contains 12 object categories. Every object category contains ten or more 3D models and more than $1,000$ real images related to the category. All those images are captured under different lighting, background complexity and contrast conditions. The dataset has both RGB images and 3D models. We used the 3D models to render corresponding depth images for the RGB images in order to train the model. We then rendered a depth image from a 3D model corresponding to each real image according to the viewpoints specified in the dataset. We randomly split the images in every category into 70% for the training set and 30% for the testing set. In order to increase the number of training samples, we apply data augmentation (DA) techniques described in the next subsection. Thus, each category has more than $10,000$ images for training the model. See Table I for more details. For all the tested 3D models, we rendered depth images using the MATLAB 3D Model Renderer [1] from multiple viewpoints by changing azimuth and elevation angles, as well as the distance between the camera and the 3D model.

B. Data Augmentation

We applied data augmentation techniques to the images contained in the PASCAL 3D+ dataset to increase the number of training samples under different conditions. Fig.8 shows the transformations applied to every input image and the corresponding rendered depth image. They were used to increase the diversity of the training dataset further.

- Scale: Every input image and its corresponding rendered depth image were randomly scaled by S $\in$ [0.5,3].
- Rotation: Every input image and its corresponding rendered depth image were randomly rotated by R $\in$ [-10,10] degrees.
- Gamma Correction: The gamma correction of each input RGB image was randomly varied by I $\in$ [0.6,2].

After applying data augmentation to the real and corresponding depth images, and using them as inputs to the model during the training process, we found that the efficiency of the network significantly improved compared to the model trained without data augmentation, even though the represented scenes were slightly warped, since they were close representations of the real images under different conditions.

[1] https://www.openu.ac.il/home/hassner/projects/poses/

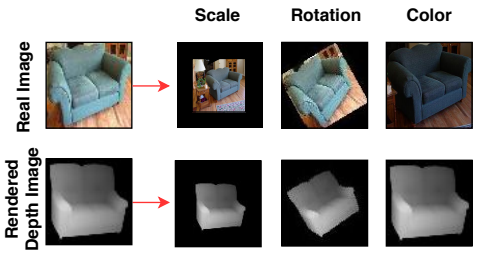

Fig. 8: Transformations (Scale, Rotation and Gamma correction) applied to every real image and its corresponding rendered depth image.

C. Parameter settings

By using data augmentation, we trained both the GAN and VPNet networks. We used the Adam optimizer [47] with $\beta_1 = 0.5$, $\beta_2 = 0.999$ and an initial learning rate of 0.0002. A batch size of 200 and $2,000$ epochs yielded the best combination. All experiments were run on a 64-bit Core I7-6700, 3.40GHz CPU with 16GB of memory, as well as one NVIDIA GTX 1080 GPU on Ubuntu 16.04 and the PyTorch [48] deep learning framework. The computational time of the proposed method for the training process takes around 2.16 minutes for each epoch with a batch-size of 64. In turn, the online estimation of depth maps and viewpoints has a performance around 3 images per second.

D. Evaluation Measures

In this work, we aim to predict the best depth image and pose viewpoint estimation from a real color image. The performance of the proposed model was evaluated in two ways:

Depth image predictions: The PASCAL 3D+ dataset is commonly used for estimating viewpoint. However, the proposed model uses it for both viewpoint estimation and depth image prediction. For depth image prediction, we used four different measures to assess the final performance. The first measure is the root mean square error (RMSE), which provides a quantitative measure of per-pixel error, computed as (8):

$$RMSE = \sqrt{\frac{1}{n}\sum_{i \in T}(B_{pred(i)} - B_{gt(i)})^2}, \qquad (8)$$

where $B_{gt(i)}$ is the real depth of pixel i, $B_{pred(i)}$ is the associated predicted depth, T is the set of valid pixels (i.e., both the ground-truth and predicted depth pixels that do not have depth values equal to zero or non-black regions as shown in Fig.8) and n is the cardinality of T.

The second measure assesses the accuracy of the proposed model to estimate errors under a given threshold, serving as an indication of how often our estimate is correct. The threshold accuracy measure from [23] is essentially the expectation that the depth value error of a given pixel in T is lower than a threshold thr^Z:

$$\delta_Z = \mathbb{E}_T[\, F(\max(\frac{B_{gt(i)}}{B_{pred(i)}}, \frac{B_{pred(i)}}{B_{gt(i)}}) < thr^Z)] \;, \qquad (9)$$

TABLE I: Number of image samples before and after data augmentation for the first six categories in the PASCAL3D+ dataset.

class	number of samples before data augmentation	number of samples after data augmentation	class	number of samples before data augmentation	number of samples after data augmentation
aero	1,681	25,215	chair	798	14,152
bike	668	12,488	table	579	13,049
boat	1,459	21,255	mbike	496	10,820
bottle	1,080	17,880	sofa	501	11,121
bus	917	16,008	train	916	16,416
car	4,637	27,822	tv	1,057	16,597
Sum	10,442	120,668	Sum	4,347	82,155

where $F(\cdot)$ represents an indicator function that yields 0 or 1. As in [23], we set $thr = 1.25$, and $Z \in \{1, 2, 3\}$.

The third measure is the Intersection Over Union (IOU) value, also referred to as the Jaccard index, computed as (10):

$$IoU = \frac{TP}{TP + FP + FN}, \tag{10}$$

where TP indicates the number of pixels whose estimated depth coincides with the real depth, FP indicates the opposite and FN indicates the number of pixels where the real depth has no predicted depth associated.

The fourth measure is the Dice score, which computes the ratio between the amount of intersection and the total number of pixels in both the prediction B_{pred} and the real depth B_{gt}, computed as (11):

$$Dice = \frac{2|B_{pred} \cap B_{gt}|}{|B_{pred}| + |B_{gt}|} = \frac{2TP}{2TP + FP + FN}. \tag{11}$$

Viewpoint predictions: To analyze the performance of the proposed viewpoint estimation method and to evaluate our model, we use two complementary evaluation measures.

The first evaluation measure is the median error [30]. The viewpoint estimation task often has predictions which are far apart: right against left or front against back. The median error (*MedErr*) is a widely used measure robust to these errors if a significant fraction of the estimates are accurate. In equation (12), we define $\triangle(B_{gt}, B_{pred})$, which measures the difference between the real depth viewpoint B_{gt} and the predicted viewpoint B_{pred}.

$$\triangle (B_{gt}, B_{pred}) = \frac{\|(B_{gt}^T B_{pred})\|_F}{\sqrt{2}}. \tag{12}$$

The second measure is the viewpoint accuracy. A small median error does not necessarily imply accurate estimates for all instances. A complementary performance measure is the fraction of instances whose predicted viewpoint is below a fixed threshold with respect to the target viewpoint. We denote this measure as Acc_θ, where θ is a threshold (e.g., $\theta = \frac{\pi}{6}$).

E. Results and Discussion

We have compared the proposed model with six alternative methods using the PASCAL3D+ dataset: [30]–[32], [34], [36], [37].

In Table II, we show the viewpoint evaluation measures for all categories of PASCAL3D+ and the different tested methods. The performance of the proposed model with GAN yielded results comparable to the alternative models. However, the accuracy of our system was superior for nine categories of PASCAL3D+: aero, with an improvement of 3%; boat, with a significant improvement of 11%; bottle and car, with a 1% improvement; chair and train, with a 5% improvement; table and mbike, with a significant improvement of 10% and 7%, respectively. However, the model proposed in [31] yielded the best accuracy for two categories: sofa and TV, with an improvement of 7% and 3% better than the proposed model, respectively. For the bus category, the model presented in [30] yielded an accuracy 2% higher than the proposed model. In turn, [32], [36] yielded an accuracy 4% higher than our results for the bike category. Globally, the proposed model yields the best mean accuracy of 89.75% among the five tested methods. In [34], the authors provided the pose estimation results for only three categories (i.e., chair, table and sofa). For the chair category, our proposed model outperformed the method in [34] with an improvement of 3.5% in MedErr, but [34] yielded better MedErr for the table category.

Regarding the median error and supporting the accuracy results, Table III shows that the proposed model yielded the lowest median error for seven categories (aero, bottle, bus, car, chair, mbike, and train) of PASCAL3D+. In addition, the proposed model yielded the lowest mean error among all tested methods.

For the tv and sofa categories, our method does not yield good results, with MedErr of 19.4 and 12.8, due to the geometric shape of these two objects. The network sometimes has a conflict, especially to estimate the correct value of the azimuth. For instance, in the example shown in Fig.9, the network can correctly estimate the depth image and the distance between the camera and the object with an error of 0.17. However, the estimation of the azimuth has an error around 30 degrees, although the estimated viewpoint is very close to the real one.

In turn, the boxplot in Fig.10 shows the accuracy values for all testing samples of the 12 categories of PASCAL3D+. For bottle and mbike, the proposed model yields a small range of values. Alternatively, the aero and boat categories yield a wider range of accuracy values with a low number of outliers. Moreover, the table and tv categories provide more than 10 outliers in the results.

As for the evaluation of the predicted depth images, we have computed the RMSE, threshold δ_Z, Dice score and IOU measures. In Table III, we show the different evaluation measures for the predicted depth images corresponding to the 12 categories of the PASCAL3D+ dataset. We have used cross-domain training to predict every depth image from a single 2D color image. As far as we are aware, there are

TABLE II: Comparison of the proposed model with current state-of-the-art algorithms for 3D pose estimation from 2D images in the PASCAL3D+ dataset under different measures. Lower is better for MedErr, and higher is better for Accuracy. The best results are highlighted in bold.

	aero	bike	boat	bottle	bus	car	chair	table	mbike	sofa	train	tv	mean
MedErr(Tulsiani and Malik, 2015) [30]	13.8	17.7	21.3	12.9	5.8	9.1	14.8	15.2	14.7	13.7	8.7	15.4	13.59
MedErr(Su et al, 2015) [32]	15.4	14.8	25.6	9.3	3.6	6.0	9.7	10.8	16.7	9.5	6.1	12.6	11.68
MedErr(Mousavian et al, 2017) [36]	13.6	**12.5**	22.8	8.3	3.1	5.8	11.9	12.5	12.3	12.8	6.3	11.9	11.15
MedErr(Grabner et al, 2018) [37]	10.0	15.6	**19.1**	8.6	3.3	5.1	13.7	11.8	12.2	13.5	6.7	**11.0**	10.88
MedErr(Mahendran et al, 2018) [31]	8.5	14.8	20.5	6.8	2.7	5.0	9.5	11.3	13.8	**9.4**	5.6	11.5	9.95
MedErr(Jogendra Nath et al, 2018) [34]	-	-	-	-	-	-	8.84	**6.00**	-	10.74	-	-	-
MedErr(Our)	**8.3**	13.2	20.7	**6.0**	**2.5**	**4.6**	**5.2**	16.5	**4.5**	12.8	**5.2**	19.4	**9.90**
$Acc_{\frac{\pi}{6}}$ (Tulsiani and Malik, 2015) [30]	0.81	0.77	0.59	0.93	**0.98**	0.89	0.80	0.62	0.88	0.82	0.80	0.80	0.8075
$Acc_{\frac{\pi}{6}}$ (Su et al, 2015) [32]	0.74	**0.83**	0.52	0.91	0.91	0.88	0.86	0.73	0.78	0.90	0.86	0.90	0.8200
$Acc_{\frac{\pi}{6}}$ (Mousavian et al, 2017) [36]	0.78	**0.83**	0.57	0.93	0.94	0.90	0.80	0.68	0.86	0.82	0.82	0.85	0.8103
$Acc_{\frac{\pi}{6}}$ (Grabner et al, 2018) [37]	0.83	0.82	0.64	0.95	0.97	0.94	0.80	0.71	0.88	0.87	0.80	0.86	0.8392
$Acc_{\frac{\pi}{6}}$ (Mahendran et al, 2018) [31]	0.87	0.82	0.64	0.97	0.97	0.95	0.92	0.68	0.85	**0.97**	0.83	**0.90**	0.8641
$Acc_{\frac{\pi}{6}}$ (Jogendra Nath et al, 2018) [34]	-	-	-	-	-	-	0.83	**0.87**	-	0.90	-	-	-
$Acc_{\frac{\pi}{6}}$ (Our)	**0.90**	0.79	**0.75**	**0.98**	0.96	**0.96**	**0.97**	0.83	**0.95**	0.90	**0.91**	0.87	**0.8975**

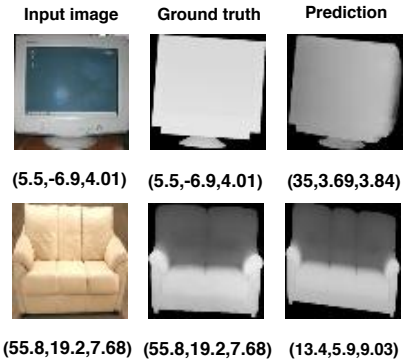

Fig. 9: Examples of the pose estimation conflict between the views of tv and sofa.

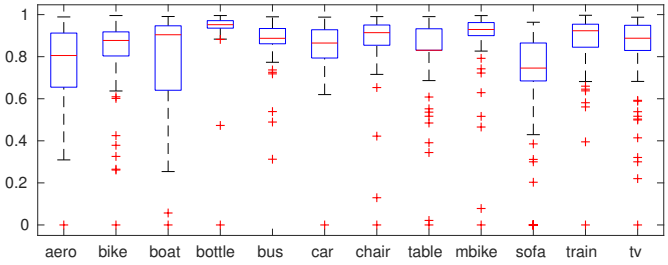

Fig. 10: Boxplot of accuracy rate for the 12 categories in PASCAL3D+ with the proposed model. Blue boxes indicate the interquartile range (Q3-Q1) of the metrics distribution. The red line inside each box represents the median value. The whiskers extend 1.5 times the length of Q1 and Q3, and (+) indicates outlier values, i.e. metrics out of the whiskers.

no alternative methods that use the PASCAL3D+ dataset for training a cross-domain model that generates depth images. Thus, we evaluate the results with two different versions of GAN and the proposed model. The first version is the GAN model proposed in [39]. The second version is the GAN model with a reconstruction loss based on the L1-norm proposed in [14]. Our model achieved the best mean results for the 12 categories with the four measures used in the evaluation. Actually, it achieved an average IOU score of 61% and Dice score of 73%. In turn, the RMSE error with the proposed model is 0.19. With $\delta_Z = 1.25$, the accuracy rate is 68%, while with $\delta_Z = 1.25^3$, the accuracy rate is increased by 16%. That shows the effect of feature matching on improving the performance of the estimation of depth images. However, the other two tested methods provided results better than our model,for mbike, sofa, train and tv.

For a qualitative assessment, Fig.11 shows how the proposed model is able to learn the features of the input images to generate the final depth images and viewpoints. The figure shows the output of the proposed model for different epochs. In addition, the performance of the proposed model for the 12 categories of PASCAL3D+ is shown in Fig.12. We show the depth image generated from a single real image against the real depth images rendered from the corresponding 3D models. We also show the three components of the estimated viewpoint and its ground-truth. These examples show that the proposed model is able to predict a depth image from the features of a single color image. In addition, the model is able to remove the image background when generating the depth images. Furthermore, the estimated viewpoints are very close to the reference ones in PASCAL3D+.

V. Conclusion

We have introduced a novel cross-domain deep model for estimating the depth image and 3D pose of the main object depicted in a 2D color image. We have designed a deep model based on two successive networks: a Generative Adversarial Network (RecDGAN) for predicting the depth images, as well as a regression CNN network for estimating the viewpoint (VPnet). The RecDGAN network consists of four sub-networks: two generators, one discriminator, and a CNN network for feature matching between the reconstructed color image and the input image. During training, the proposed model is fed with a single 2D image of an object and the corresponding depth image rendered from a 3D model of the same object. The generated depth image is fed into VPnet to estimate the viewpoint. The model performance has been evaluated on the PASCAL3D+ dataset, yielding promising results

TABLE III: Results for depth image estimation from 2D color images on the PASCAL3D+ dataset under different measures with (a) GAN proposed in [39], (b) GAN with a reconstruction loss proposed in [14] and (c) the proposed model. Lower is better for the RMSE metric, and higher is better for the other measures. The best results are highlighted in bold.

		aero	bike	boat	bottle	bus	car	chair	table	mbike	sofa	train	tv	mean
GAN Model	IoU	0.46	0.26	0.50	0.80	0.62	0.71	0.42	0.44	0.48	0.61	0.56	0.78	0.55
	Dice	0.62	0.40	0.66	0.87	0.76	0.82	0.57	0.60	0.61	0.73	0.71	0.87	0.69
	RMSE (linear)	0.21	0.26	**0.20**	**0.14**	0.16	0.18	0.23	0.23	0.24	**0.15**	0.19	0.15	0.20
	threshold δ<1.25	0.77	0.53	**0.69**	0.66	0.47	0.63	0.71	0.75	0.65	0.59	0.56	0.53	0.63
	threshold δ<1.25^2	0.83	0.60	**0.78**	0.83	0.63	0.74	0.81	0.81	0.71	0.78	0.68	0.69	0.74
	threshold δ<1.25^3	0.86	0.65	**0.84**	0.89	0.73	0.83	0.85	0.83	0.77	0.87	0.75	0.81	0.81
GAN with a reconstruction loss	IoU	0.49	0.32	0.51	0.79	0.67	0.73	0.47	0.42	0.51	**0.63**	**0.61**	**0.80**	0.58
	Dice	0.64	0.41	0.66	0.88	0.79	0.84	0.62	0.58	0.67	**0.76**	**0.75**	**0.89**	0.71
	RMSE (linear)	0.20	0.24	**0.20**	0.17	**0.15**	0.18	0.23	0.23	**0.22**	0.18	**0.18**	0.16	0.20
	threshold δ<1.25	**0.83**	0.65	0.68	**0.76**	**0.61**	0.67	0.72	0.77	**0.72**	**0.66**	**0.62**	**0.54**	**0.69**
	threshold δ<1.25^2	**0.87**	0.68	0.76	0.85	**0.75**	0.79	0.80	**0.84**	**0.80**	**0.80**	**0.69**	**0.71**	**0.78**
	threshold δ<1.25^3	**0.89**	0.70	0.82	0.89	**0.80**	0.85	0.83	0.85	**0.84**	**0.88**	**0.79**	**0.82**	0.83
Our Model	IoU	**0.52**	**0.43**	**0.56**	**0.82**	**0.70**	**0.75**	**0.52**	**0.49**	**0.53**	0.62	0.54	0.78	**0.61**
	Dice	**0.66**	**0.55**	**0.71**	**0.90**	**0.81**	**0.86**	**0.67**	**0.65**	**0.68**	0.75	0.69	0.87	**0.73**
	RMSE (linear)	**0.18**	**0.23**	**0.20**	**0.14**	**0.15**	**0.16**	**0.21**	**0.22**	0.23	**0.16**	0.20	**0.14**	**0.19**
	threshold δ<1.25	0.80	**0.70**	0.65	**0.76**	0.58	**0.69**	**0.74**	**0.78**	0.71	0.61	0.58	0.52	0.68
	threshold δ<1.25^2	0.85	**0.74**	0.75	**0.86**	0.72	**0.82**	**0.82**	**0.84**	0.79	0.79	0.68	0.70	**0.78**
	threshold δ<1.25^3	0.87	**0.76**	0.82	**0.91**	0.79	**0.87**	**0.86**	**0.87**	**0.84**	0.86	0.76	0.81	**0.84**

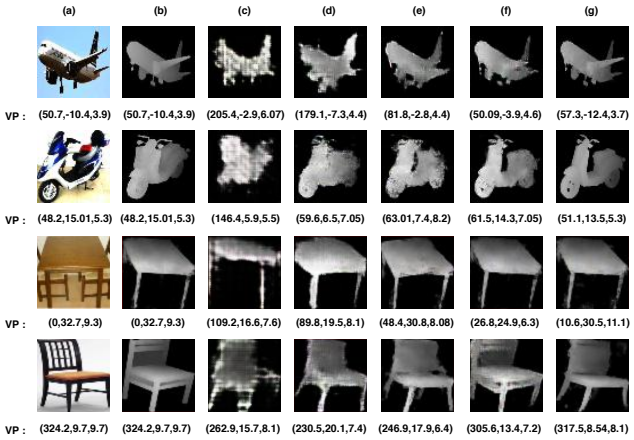

Fig. 11: Example of depth predictions with our model. For each image, we show (a) input image, (b) ground truth, (c) output at epoch 100, (d) output at epoch 400, (e) output at epoch 1000, (f) output at epoch 1500, (g) output at epoch 2000 (final generated depth image). All images with the corresponding estimated viewpoints (VP) including (Azimuth and Elevation angles and Distance between object and camera). More results with the proposed model are given in the supplementary material.

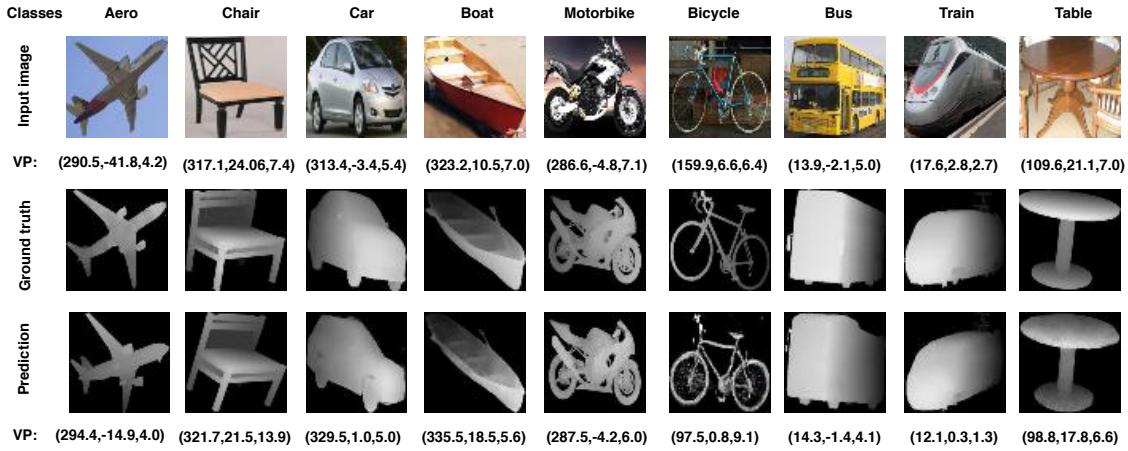

Fig. 12: Input images with the labeled viewpoints and corresponding depth images rendered from the associated 3D models of all categories of PASCAL 3D+, and generated depth images with the estimated viewpoints. More details about the performance of the proposed model are given in the supplementary material.

with a high precision rate and an acceptable computational cost for predicting depth images and viewpoints from input color images. Future work aims at applying the proposed pose estimation to an object grasping framework based on a single RGB camera.

REFERENCES

[1] I. Haritaoglu, D. Harwood, and L. S. Davis, "W 4 s: A real-time system for detecting and tracking people in 2 1/2d," in *European Conference on computer vision*. Springer, 1998, pp. 877–892.

[2] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "Stargan: Unified generative adversarial networks for multi-domain image-to-image translation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8789–8797.

[3] N. Wang, Y. Zhang, Z. Li, Y. Fu, W. Liu, and Y.-G. Jiang, "Pixel2mesh: Generating 3d mesh models from single rgb images," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 52–67.

[4] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.

[5] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[6] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," *arXiv preprint*, 2017.

[7] M. Zhang, R. Wang, X. Gao, J. Li, and D. Tao, "Dual-transfer face sketch–photo synthesis," *IEEE Transactions on Image Processing*, vol. 28, no. 2, pp. 642–657, 2018.

[8] L. Ge, H. Liang, J. Yuan, and D. Thalmann, "Robust 3d hand pose estimation in single depth images: from single-view cnn to multi-view cnns," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3593–3601.

[9] D. Mehta, H. Rhodin, D. Casas, P. Fua, O. Sotnychenko, W. Xu, and C. Theobalt, "Monocular 3d human pose estimation in the wild using improved cnn supervision," in *2017 International Conference on 3D Vision (3DV)*. IEEE, 2017, pp. 506–516.

[10] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," in *Advances in neural information processing systems*, 2014, pp. 2366–2374.

[11] L. Ge, H. Liang, J. Yuan, and D. Thalmann, "3d convolutional neural networks for efficient and robust hand pose estimation from single depth images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1991–2000.

[12] A. Kadambi, A. Bhandari, and R. Raskar, "3d depth cameras in vision: Benefits and limitations of the hardware," in *Computer Vision and Machine Learning with RGB-D Sensors*. Springer, 2014, pp. 3–26.

[13] D. Tao, J. Cheng, Z. Yu, K. Yue, and L. Wang, "Domain-weighted majority voting for crowdsourcing," *IEEE transactions on neural networks and learning systems*, vol. 30, no. 1, pp. 163–174, 2018.

[14] T. Kim, M. Cha, H. Kim, J. Lee, and J. Kim, "Learning to discover cross-domain relations with generative adversarial networks," *arXiv preprint arXiv:1703.05192*, 2017.

[15] D. Jiang, Y. Hu, S. Yan, L. Zhang, H. Zhang, and W. Gao, "Efficient 3d reconstruction for face recognition," *Pattern Recognition*, vol. 38, no. 6, pp. 787–798, 2005.

[16] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recognition: A literature survey," *ACM computing surveys (CSUR)*, vol. 35, no. 4, pp. 399–458, 2003.

[17] P. V. Harman, J. Flack, S. Fox, and M. Dowley, "Rapid 2d-to-3d conversion," in *Stereoscopic Displays and Virtual Reality Systems IX*, vol. 4660. International Society for Optics and Photonics, 2002, pp. 78–87.

[18] A. Saxena, S. H. Chung, and A. Y. Ng, "Learning depth from single monocular images," in *Advances in neural information processing systems*, 2006, pp. 1161–1168.

[19] A. Saxena, M. Sun, and A. Y. Ng, "Make3d: Depth perception from a single still image." in *AAAI*, 2008, pp. 1571–1576.

[20] K. Clayden, "Personality, motivation and level of involvement of land-based recreationists in the irish uplands," Ph.D. dissertation, Waterford Institute of Technology, 2012.

[21] B. Li, C. Shen, Y. Dai, A. Van Den Hengel, and M. He, "Depth and surface normal estimation from monocular images using regression on deep features and hierarchical crfs," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1119–1127.

[22] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "Slic superpixels compared to state-of-the-art superpixel methods," *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 11, pp. 2274–2282, 2012.

[23] F. Liu, C. Shen, and G. Lin, "Deep convolutional neural fields for depth estimation from a single image," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5162–5170.

[24] S. Zhang, N. Li, C. Qiu, Z. Yu, H. Zheng, and B. Zheng, "Depth map prediction from a single image with generative adversarial nets," *Multimedia Tools and Applications*, pp. 1–18, 2018.

[25] T. Plötz and S. Roth, "Automatic registration of images to untextured geometry using average shading gradients," *International Journal of Computer Vision*, vol. 125, no. 1-3, pp. 65–81, 2017.

[26] H. A. Rashwan, S. Chambon, P. Gurdjos, G. Morin, and V. Charvillat, "Towards multi-scale feature detection repeatable over intensity and depth images," in *Image Processing (ICIP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 36–40.

[27] H. A. Rashwan, S. Chambon, P. Gurdjos, G. Morin, and V. Charvillat, "Using curvilinear features in focus for registering a single image to a 3d object supplemental materials," *IEEE Transactions on Image Processing*, pp. 1–1, 2019.

[28] S. Abdulwahab., H. A. Rashwan., J. Cristiano., S. Chambon., and D. Puig., "Effective 2d/3d registration using curvilinear saliency features and multi-class svm," in *Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 5: VISAPP,*, INSTICC. SciTePress, 2019, pp. 354–361.

[29] D. Eigen and R. Fergus, "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 2650–2658.

[30] S. Tulsiani and J. Malik, "Viewpoints and keypoints," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1510–1519.

[31] S. Mahendran, M. Y. Lu, H. Ali, and R. Vidal, "Monocular object orientation estimation using riemannian regression and classification networks," *arXiv preprint arXiv:1807.07226*, 2018.

[32] H. Su, C. R. Qi, Y. Li, and L. J. Guibas, "Render for cnn: Viewpoint estimation in images using cnns trained with rendered 3d model views," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 2686–2694.

[33] C. Zimmermann and T. Brox, "Learning to estimate 3d hand pose from single rgb images," in *International Conference on Computer Vision*, vol. 1, no. 2, 2017, p. 3.

[34] J. Nath Kundu, A. Ganeshan, and R. Venkatesh Babu, "Object pose estimation from monocular image using multi-view keypoint correspondence," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 0–0.

[35] Y. Gao and A. L. Yuille, "Estimation of 3d category-specific object structure: Symmetry, manhattan and/or multiple images," *International Journal of Computer Vision*, vol. 127, no. 10, pp. 1501–1526, 2019.

[36] A. Mousavian, D. Anguelov, J. Flynn, and J. Košecká, "3d bounding box estimation using deep learning and geometry," in *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*. IEEE, 2017, pp. 5632–5640.

[37] A. Grabner, P. M. Roth, and V. Lepetit, "3d pose estimation and 3d model retrieval for objects in the wild," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3022–3031.

[38] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," *arXiv preprint*, 2017.

[39] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[40] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su *et al.*, "Shapenet: An information-rich 3d model repository," *arXiv preprint arXiv:1512.03012*, 2015.

[41] L. Yu, W. Zhang, J. Wang, and Y. Yu, "Seqgan: Sequence generative adversarial nets with policy gradient." in *AAAI*, 2017, pp. 2852–2858.

[42] A. Brock, T. Lim, J. M. Ritchie, and N. Weston, "Neural photo editing with introspective adversarial networks," *arXiv preprint arXiv:1609.07093*, 2016.
[43] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. P. Aitken, A. Tejani, J. Totz, Z. Wang *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network." in *CVPR*, vol. 2, no. 3, 2017, p. 4.
[44] C. K. Sønderby, J. Caballero, L. Theis, W. Shi, and F. Huszár, "Amortised map inference for image super-resolution," *arXiv preprint arXiv:1610.04490*, 2016.
[45] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. icml*, vol. 30, no. 1, 2013, p. 3.
[46] Y. Xiang, R. Mottaghi, and S. Savarese, "Beyond pascal: A benchmark for 3d object detection in the wild," in *Applications of Computer Vision (WACV), 2014 IEEE Winter Conference on*. IEEE, 2014, pp. 75–82.
[47] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
[48] A. Paszke, S. Gross, S. Chintala, and G. Chanan, "Pytorch: Tensors and dynamic neural networks in python with strong gpu acceleration," 2017.

Saddam Abulwahab received the BS degree in Computer Science from Hodeidah University (Hodeidah, Yemen) in 2012, the M.Sc. Degree in Computer Security And Artificial Intelligence from URV (Tarragona, Spain) in 2017. Between 2012 and 2016, he joined the Department of Computer Science and Engineering at Hodeidah University in Yemen as a lecturer. In 2016, he joined the Intelligent Technologies for Advanced Knowledge Acquisition ITAKA Group, DEIM at URV (Tarragona, Spain), where he was a Master Student until Sep. 2017. From Sep. 2017 until now, he joined the IRCV Group where he is a Ph.D. Student. His research interests include image processing, computer vision, machine learning and pattern recognition.

Hatem A. Rashwan received the B.S. and M.S. degrees in electrical engineering from South Valley University (Egypt) in 2002, 2007. He received the PhD degree in Computer Vision from Rovira i Virgili University in 2014. Between 2004 and 2009, he joined the Electrical Engineering Department, South Valley University as an Assistant Lectur. From Jan. 2010 until Oct. 2014, he joined IRCV Group, Department of Computer Engineering and Mathematics at Rovira i Virgili University (Spain) as a PhD student and research assistant. From Nov. 2014 until August 2017, he is a PostDoc in the VORTEX group, IRIT, CNRS, INP-Toulouse, University of Toulouse (France). From 2018 until now, he is a Beatriu de Pins researcher in URV. His research interests include image processing, computer vision, machine learning and pattern recognition.

Miguel ngel Garca received the B.S., M.S., and Ph.D. degrees in computer science from the Polytechnic University of Catalonia (Barcelona, Spain) in 1989, 1991, and 1996, respectively. He joined the Department of Software at the Polytechnic University of Catalonia in 1996 as an Assistant Professor. From 1997 to 2006, he was with the Department of Computer Science and Mathematics at Rovira i Virgili University (Tarragona, Spain), where he was the Head of Intelligent Robotics and Computer Vision group. In 2006, he joined the Department of Informatics Engineering at Autonomous University of Madrid (Spain), where he is currently Associate Professor. His research interests include mobile robotics, image processing, and 3-D modeling.

Mohammed Jabreel Mohammed Jabreel has received a Bachelor's degree in Computer Science in 2009 from the Hodiedah University, Yemen and a Master's degree from the Universitat Rovira i Virgili (Catalonia, Spain) in 2015. He is currently in his last year of PhD in Computer Engineering at the Department of Computer Science and Mathematics, Universitat Rovira i Virgili. His research interests concern Natural Language Processing, Opinion Mining and their application in real case studies. Currently, he started to work on developing Deep Learning-based systems for Cross-Domains and their applications to the Natural Language Processing, Sentiment Analysis and Computer Vision fields.

Sylvie Chambon received the Ph.D. degree in computer science from the University of Toulouse, Toulouse, France, working on Colour stereoscopic matching with occlusions.,From 2006 to 2007, she was a Postdoctoral Researcher with Tlcom Paris, working on in multimodal registration of medical images. From 2008 to 2011, she was a Permanent Researcher with IFSTTAR (the French institute of science and technology for transport, development and networks), working on segmentation of thin structures and, in particular, road cracks. Since September 2011, she has been an Assistant Professor with the Institut de Recherche en Informatique de Toulouse (IRIT), INP-ENSEEIHT, Universit of Toulouse. Her research interest includes matching, feature detection and tracking, and segmentation of thin structures and urban scenes.

Domenec Puig received the M.S. and Ph.D. degrees in computer science from the Polytechnic University of Catalonia (Barcelona, Spain) in 1992 and 2004, respectively. In 1992, he joined the Department of Computer Engineering and Mathematics at Rovira i Virgili University (Tarragona, Spain), where he is currently Professor. Since July 2006, he is the Head of the Intelligent Robotics and Computer Vision group at the same university. His research interests include image processing, texture analysis, perceptual models for image analysis, scene analysis, and mobile robotics.