Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210159

Promising Depth Map Prediction Method from a Single Image Based on Conditional Generative Adversarial Network

Saddam ABDULWAHAB [a,1], Hatem A. RASHWAN [a], Armin MASOUMIAN [a], Najwa SHARAF [a] and Domenec PUIG [a]

[a] *DEIM, Universitat Rovira i Virgili, 43003 Tarragona, Spain*

Abstract. Pose estimation is typically performed through 3D images. In contrast, estimating the pose from a single RGB image is still a difficult task. RGB images do not only represent objects' shape, but also represent the intensity that is relative to the viewpoint, texture, and lighting condition. While the 3D pose estimation from depth images is considered a promising approach since the depth image only represents objects' shape. Thus, it is necessary to know what is the appropriate method that can be used for predicting the depth image from a 2D RGB image and then to use for getting the 3D pose estimation. In this paper, we propose a promising approach based on a deep learning model for depth estimation in order to improve the 3D pose estimation. The proposed model consists of two successive networks. The first network is an autoencoder network that maps from the RGB domain to the depth domain. The second network is a discriminator network that compares a real depth image to a generated depth image to support the first network to generate an accurate depth image. In this work, we do not use real depth images corresponding to the input color images. Our contribution is to use 3D CAD models corresponding to objects appearing in color images to render depth images from different viewpoints. These rendered images are then used as ground truth and to guide the autoencoder network to learn the mapping from the image domain to the depth domain. The proposed model outperforms state-of-the-art models on the publicly PASCAL 3D+ dataset.

Keywords. Deep learning, depth prediction, image segmentation, UNet, UNet++, image to image translation.

1. Introduction

Inferring depth and 3D pose estimation from a single RGB image is one of the most challenging problems in computer vision. In this work, we address the challenging problem of depth estimation that is useful to determine pose estimation showing in a scene using a monocular camera. Practicality, the important challenge in a 3D pose estimation directly from a single image is the ambiguity of the object shape in a monocular image. Since 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

[1] Corresponding Author: Saddam Abdulwahab. E-mail:saddam.abdulwahab@gmail.com

acquisition (e.g., camera pose and gamma correction conditions). Thus, estimating a 3D pose requires the depth image that only contains the shapes of the objects in order to estimate the correct pose of the main objects in a 3D scene.

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, [1] presents a framework for depth and surface normal estimation from 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 [2] that is used to segment the depth images into super-pixels). [1] used then refined 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 estimated depth map. In [3], a three-layer CNN trained with a per-pixel Euclidean loss was presented to convert the given color image to a geometrically meaningful output image. Besides, they used Conditional Random Fields (CRF) as a loss layer to enforce local consistency in the output image.

This work is close in spirit to that of [4,5,6] in the sense that we also use a deep learning approach to retrieve depth maps from a single image. Our method is a promising method since it can be applied to predict depth images for indoor and outdoor scenarios. Besides, it can be used as a co-representation method to be applied to predict the pose estimation from a single RGB image. Furthermore, it is producing promising results with a high precision rate and an acceptable computational cost.

In this work, we propose to use an autoencoder network as a generator based on UNet and UNet++ models [7,8]. In particular, a cutting-edge technique for image transformation as a baseline network for predicting a depth image from a single color image. However, with the lack of annotated training data for depth images of objects, we use 3D CAD models for rendering depth images from different viewpoints. The obtained depth images are used to train the autoencoder network. The proposed model consists of two successive networks. The first network is depth estimation that learns to map the RGB image domain into the depth image domain. In order to enforce the generator to generate a depth close to the ground truth, we propose a second network is a discriminator network that helps the first network by comparing the ground truth and generated depth images. The two networks are integrated into a single pipeline to solve the problem of depth image estimation. Figure 1 shows the proposed framework for depth estimation from a single image using technique segmentation.

To the best of our knowledge, this work is the first attempt to use an autoencoder network used for the image segmentation purpose as a generator to a training model for estimating depth maps of the main object depicted in a 2D image. Consequently, the main contributions of this paper are the following:

- We propose an autoencoder segmentation network as a generator that can predict a depth image from a single 2D color image of an object.
- We propose a discriminator network to achieve a more accurate comparison of the ground truth and generated depth to enforce the autoencoder network to generate an accurate dense depth image.
- The integration of the two networks into a single pipeline to solve the problems of generating a depth image from a single color image.

This paper is organized as follows. Section 2 describes the proposed methodology to estimate a depth image using segmentation. Section 3 describes experimental results. Finally, Section 4 concludes this work and suggests future lines of research.

2. Methodology

This section explains the proposed scheme, the tools, and the resources being used in this work. We formulate the problem in subsection 2.1. The remaining subsections explain each part of the proposed model in detail.

2.1. Problem Formulation

Let $a \in A$ be a 2D color image, and the problem of generating its corresponding depth image, $b \in B$, can be dened formally as a function $f : A \rightarrow B$ maps elements from domain A to ones in its co-domain B. Figure 1 shows the graphical description of the system. It contains two main modules. The first one is a depth generator G based on an autoencoder segmentation Network, and the second one is the discriminator network D based on a CNN.

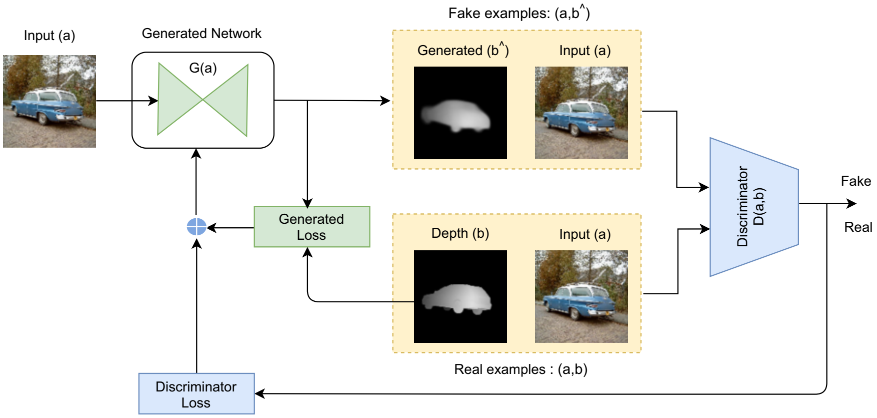

Figure 1. General overview of the proposed depth estimation model.

2.2. Generator Network

Two main variations of our autoencoder segmentation network are proposed in this paper as a generator network. Both of them are encoder-decoder neural network architecture. The first network is UNet [7], it involves convolution layers, and it does not include a fully connected layer that is demanding on a big amount of data. This network is simple, efficient, and easily used. It consists of two parts: the first one is an encoder that obtains different image feature levels continuously sampled through multiple convolution layers. Also, we tested the UNet++ [8], which consists of a series of nested dense convolutional blocks, as a encoder to choose the best between UNet and UNet++ networks

The second one is a decoder that performs multi-layer deconvolution on the top-level feature map and combines different feature levels in the down-sampling process to restore the feature map to the original input image size and completes the end-to-end depth estimation task from the input image. Besides, it uses the skip connection operation to connect each pair of down-sampling layers and the up-sampling layer that makes the spatial information directly applied to much deeper layers and a more accurate segmentation result.

The generator G learns the mapping from an input color image to the corresponding depth image. The input to the segmentation network is a 2D color image, a, and it generates a depth image, $\hat{b}$.

In order to optimize the structural similarity between the depth image and ground truth, we use two loss functions: the first one is a *MSE* loss function based on feature matching that can be defined as follows (1):

$$\mathfrak{L}_{\mathfrak{gan}}(a,b,G(a)) = \frac{1}{n}\sum_{i\in T}^{n} f(\hat{b}_{(i)} - b_{(i)})^2, \tag{1}$$

where a is input 2D color image, G is a generator network, f is the *MSE* error, $b_{(i)}$ is the real depth of pixel i, $\hat{b}_{(i)}$ is the associated predicted depth by generator network, 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 Figure 2 and n is the cardinality of T.

2.3. Discriminator Network

The generator network generates depth images b that belong to domain B from the domain A of color images. To model this additional constraint, we proposed a discriminator network that 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 the first convolutional layer C_{n1} followed by an output logistic unit*LeakyReLU* [3,9]. The idea of our approach is to train the generator to generate samples very close to the real samples and the samples have to be in the depth image domain. To model this additional constraint, we train a discriminator neural network D to distinguish between a real sample consisting of (input color image and real depth image rendered from 3D CAD models) and a fake sample consisting of(input image and generated depth image from the generator G). The discriminator network D is used to determine whether the depth images estimated by the generator G are comparable to depth images or not.

In addition, it provides a second loss measure, along with the reconstruction error of the generated depth map, that is useful for training an accurate generator to generate a dense depth image and minimize the difference between the corresponding features and avoid the over-fitting and make the training more stable and converge faster. The discriminator is trained by minimizing the following binary cross-entropy (BCE) loss is defined as follows (2):

$$\ell_{Dis}(D,a,b,\hat{b}) = -\mathbb{E}_{a\hat{b}b}[log(D(a,b)) + log(1 - D(a,\hat{b})] \tag{2}$$

2.4. Total Loss

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

$$\mathfrak{L}(G,D,a,b,\hat{b}) = \ell_{gan}(G,D,a,G(a)) + \ell_{Dis}(D,a,b,\hat{b}) \tag{3}$$

This loss $\mathfrak{L}(G,D,a,b,\hat{b})$ is efficiently integrated into the back-propagation for the generator network through ADAM optimization.

3. Experiments and Results

This section describes the experiments performed to evaluate the proposed model on the publicly PASCAL 3D+ dataset using various evaluation measures.

3.1. Dataset

In this work, a comprehensive set of experiments have been conducted to validate the performance of the proposed model on the public PASCAL3D+ dataset [10], which contains 12 object categories. Every object category contains ten or more 3D models and more than $1,000$ color images related to every category. We used the 3D models to render corresponding depth images for the RGB images to train the proposed model. We render a depth image from a 3D model corresponding to each color image according to the viewpoints specified in the dataset. We randomly split the images in each category into 70% for the training set and 30% for the testing set. To increase the number of training samples, we apply data augmentation (DA) techniques Shown in Figure2 that show the transformations applied to every input image and the corresponding depth images. Thus, each category has more than $10,000$ images for training the model. After applying data augmentation to the real color images and corresponding depth ones, and using them as inputs to the model for the training process, we found that the efficiency of the network significantly improved due to exposing the model with more difficult samples and samples under different conditions. For all the tested 3D models, we rendered depth images using the MATLAB 3D Model Renderer [2] based on the viewpoints (i.e, azimuth and elevation angles), as well as the distance between the camera and the 3D model obtained from the annotation of the PASCAL 3D+ dataset.

3.2. Parameter settings

We used the Adam optimizer [11] with $\beta_1 = 0.5$, $\beta_2 = 0.999$ and an initial learning rate of 0.0001. A batch size of 4 with 1000 epochs yielded the best combination. The input images is reshaped to 128×128 pixels and normalizes through divided by 255. For all these experiments, we used a 64-bit I7-6700, 3.40GHz CPU with 16GB of memory, as well as one NVIDIA GTX 1080 GPU on Ubuntu 16.04. We used the Pytorch [12] deep

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

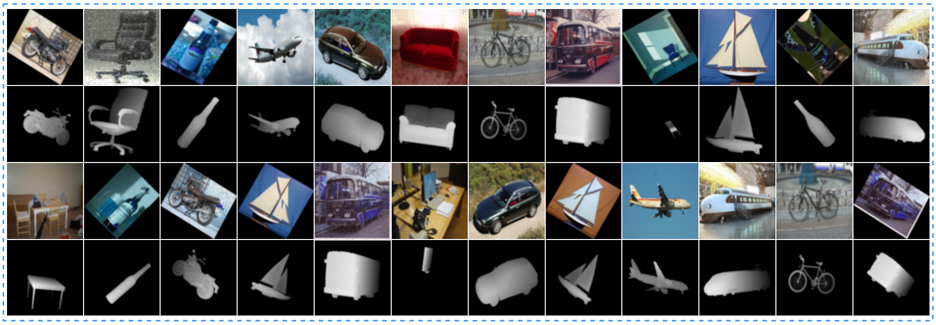

Figure 2. Transformations (flipping, blurring, noise, and rotation) are applied to every real image and its corresponding rendered depth image in all transformations, except blurring and noise, we apply them for the real image only.

learning framework. The computational time of the proposed method for the training process takes around 1.2 minutes for each epoch with a batch size of 4. In turn, the online estimation of depth maps has a performance of around 7 images per second.

3.3. Evaluation Measures

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 (4):

$$RMSE = \sqrt{\frac{1}{n}\sum_{i\in T}(\hat{b}_{(i)} - b_{(i)})^2}, \tag{4}$$

where $b_{(i)}$ is the real depth of pixel i, $\hat{b}_{(i)}$ 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 Figure 2 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 [3] is essentially the expectation that the depth value error of a given pixel in T is lower than a threshold thr^Z:

$$\delta_Z = \mathbb{E}_T[F(\max(\frac{b_{(i)}}{\hat{b}_{(i)}}, \frac{\hat{b}_{(i)}}{b_{(i)}}) < thr^Z)] , \tag{5}$$

where $F(\cdot)$ represents an indicator function that yields 0 or 1. As in [3], 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 that can be computed as (6):

$$IoU = \frac{TP}{TP+FP+FN}, \tag{6}$$

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 predicted $\hat{b}$ and the real depth b that can be defined as (7):

$$Dice = \frac{2|\hat{b} \cap b|}{|\hat{b}| + |b|} = \frac{2TP}{2TP + FP + FN}. \tag{7}$$

Table 1. Results for depth image estimation from 2D color images on the PASCAL3D+ dataset under different measures with (a) GAN proposed in [13], (b) GAN with a reconstruction loss proposed in [14], (c) Adversarial Learning proposed in [6] and (d) 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 $\delta < 1.25$	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 $\delta < 1.25^2$	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 $\delta < 1.25^3$	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		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 $\delta < 1.25$	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 $\delta < 1.25^2$	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 $\delta < 1.25^3$	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
Adversarial Learning		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 $\delta < 1.25$	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 $\delta < 1.25^2$	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 $\delta < 1.25^3$	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
Our Model	UNet	IoU	**0.53**	0.41	**0.61**	0.77	0.75	0.79	**0.62**	**0.53**	0.55	**0.70**	0.65	**0.78**	**0.64**
		Dice	0.69	0.51	**0.75**	0.86	**0.83**	**0.88**	**0.75**	**0.68**	0.69	**0.82**	0.77	**0.87**	**0.758**
		RMSE (linear)	**0.18**	0.25	**0.17**	**0.11**	**0.15**	**0.14**	**0.20**	**0.21**	0.22	0.20	0.17	0.17	**0.18**
		threshold $\delta < 1.25$	**0.87**	0.70	0.67	**0.82**	0.65	**0.78**	**0.78**	**0.82**	0.81	**0.84**	0.72	**0.69**	0.762
		threshold $\delta < 1.25^2$	**0.89**	0.74	0.76	**0.87**	0.78	**0.86**	**0.87**	**0.85**	**0.86**	**0.89**	0.76	**0.76**	0.824
		threshold $\delta < 1.25^3$	**0.90**	**0.78**	**0.85**	**0.93**	0.85	0.88	**0.89**	**0.87**	0.88	**0.91**	0.81	**0.86**	0.867
	UNet++	IoU	**0.53**	0.42	**0.61**	0.77	**0.76**	**0.81**	0.61	0.52	**0.56**	0.69	**0.67**	0.76	**0.64**
		Dice	**0.70**	0.51	**0.75**	0.86	**0.83**	**0.88**	**0.75**	0.67	**0.70**	0.81	**0.78**	0.85	0.757
		RMSE (linear)	**0.18**	0.24	**0.17**	0.12	**0.15**	**0.14**	**0.20**	**0.21**	**0.21**	0.21	**0.16**	0.17	**0.18**
		threshold $\delta < 1.25$	**0.87**	**0.71**	**0.69**	**0.82**	**0.67**	**0.78**	**0.78**	0.81	**0.82**	0.83	**0.73**	0.67	**0.765**
		threshold $\delta < 1.25^2$	**0.89**	**0.75**	**0.78**	**0.87**	**0.79**	**0.86**	**0.87**	0.84	**0.86**	**0.89**	**0.78**	**0.76**	**0.828**
		threshold $\delta < 1.25^3$	**0.90**	**0.78**	**0.85**	**0.93**	**0.86**	**0.89**	**0.89**	0.86	**0.89**	**0.91**	**0.83**	**0.86**	**0.87**

3.4. Results and Discussion

We have compared the proposed model with three alternative methods using the PASCAL3D+ dataset: [13,14,6]. In Table 1, we show the four evaluation measures for the predicted depth images corresponding to the 12 categories of the PASCAL3D+ dataset. We evaluate the results with three different versions of GAN and our proposed model. The first version is the GAN model proposed in [13]. The second version is the GAN model with a reconstruction loss based on the L1-norm proposed in [14]. The third version is the adversarial learning model proposed in [6]. Our model achieved the best mean results for the 12 categories with the four measures used in the evaluation. It achieved an average IOU score of 64% and a Dice score of 75.8%. In turn, the RMSE error with the proposed model is 0.18. With $\delta_Z = 1.25$, the accuracy rate is 76.5%, while with $\delta_Z = 1.25^3$, the accuracy rate is increased by 3%. That shows the effect of discriminator and feature matching on improving the performance of the estimation of depth images. However, the other three tested methods provided results better than our model, for bike, and bottle.

For a qualitative assessment, Figure 3 shows how the proposed model can generate depth images that are very close to the ground truth. The figure shows the output of the proposed model for different categories of PASCAL 3D+.

In addition, the performance of the proposed model for some of the categories of PASCAL 3D+ is shown in Figure 4. We show the depth image generated against the real depth images rendered from the corresponding 3D models. Besides, we show composite images from the color and the generated depth image in (row 1 and row 2). These examples show that the proposed model can predict a proper depth image that has the pose of the object in color images.

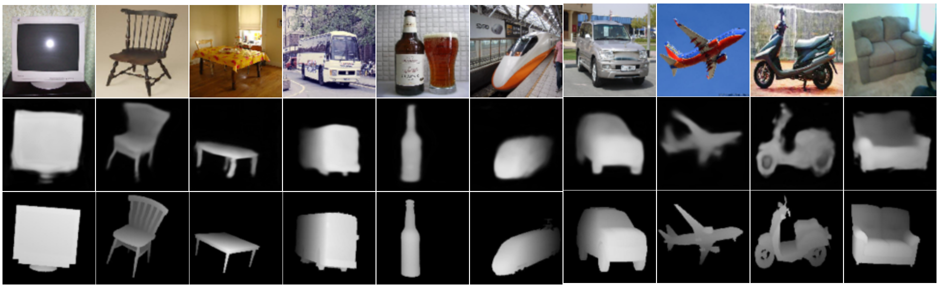

Figure 3. Intensity images (row 1), resulting depth images (row 2), Ground-truth depth images (row 3).

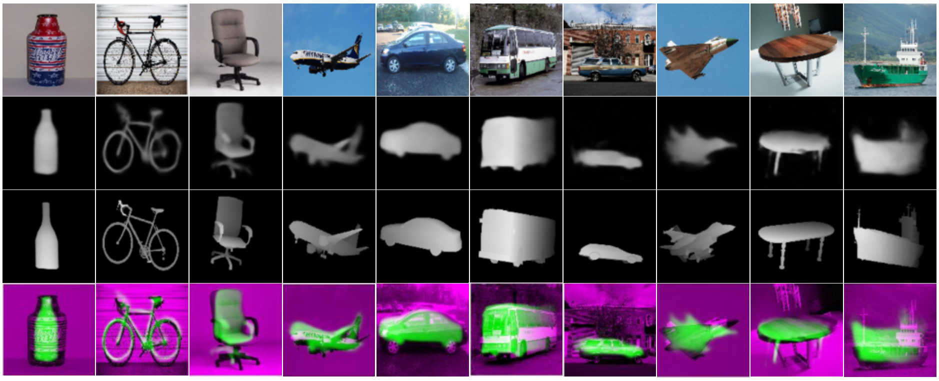

Figure 4. We show some correct and erroneous predictions given by our final method compared to the ground-truth. As it is shown, in the first six columns, we show the correct prediction, and in the last four-columns, we show the error prediction. Intensity images (row 1), resulting depth images (row 2), Ground-truth depth images (row 3), and composite images from intensity and resulting depth images (row 4).

4. Conclusion

We have introduced a novel cross-domain deep model for estimating a depth image of the main object depicted in a 2D color image. We have designed a deep model based on two deep networks. The first network is an autoencoder segmentation network, called a generator. The generator network maps the color image to a depth image. The second network is a discriminator network to achieve more comparison and allows the system to generate a dense depth image. During training, the proposed model in the first network is fed with a single 2D image for the object and the corresponding depth image rendered from a 3D model of the same object. Both the input color image and the depth image generated by the generator network are fed into the discriminator to make a more accurate comparison to the ground truth images to help in generating a more precise depth image. The model performance has been evaluated on the PASCAL3D+ dataset, yielding promising results with a high precision rate and a low computational cost comparing to the state of the art. Future work aims at applying the generated depth maps to predict the pose estimation of objects showing in a scene.

5. Acknowledgements

Financial support was given by the pre-doctoral grant (FI 2020) given by the Catalan government.

References

[1] 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*, pp. 1119–1127, 2015.

[2] 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.

[3] 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*, pp. 5162–5170, 2015.

[4] 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*, pp. 2366–2374, 2014.

[5] D. PUIG, "Mgnet: Depth map prediction from a single photograph using a multi-generative network," in *Artificial Intelligence Research and Development: Proceedings of the 22nd International Conference of the Catalan Association for Artificial Intelligence*, vol. 319, p. 356, IOS Press, 2019.

[6] S. Abdulwahab, H. A. Rashwan, M. A. Garcia, M. Jabreel, S. Chambon, and D. Puig, "Adversarial learning for depth and viewpoint estimation from a single image," *IEEE Transactions on Circuits and Systems for Video Technology*, 2020.

[7] 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*, pp. 234–241, Springer, 2015.

[8] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, "Unet++: A nested u-net architecture for medical image segmentation," in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, pp. 3–11, Springer, 2018.

[9] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. icml*, vol. 30, p. 3, 2013.

[10] 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*, pp. 75–82, IEEE, 2014.

[11] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.

[12] A. Paszke, S. Gross, S. Chintala, and G. Chanan, “Pytorch: Tensors and dynamic neural networks in python with strong gpu acceleration,” 2017.

[13] 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*, pp. 2672–2680, 2014.

[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.