IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received October 13, 2021, accepted December 15, 2021, date of publication December 23, 2021, date of current version January 18, 2022.

Digital Object Identifier 10.1109/ACCESS.2021.3137797

Analysis of Depth and Semantic Mask for Perceiving a Physical Environment Using Virtual Samples Generated by a GAN

JAVIER MALDONADO-ROMO[1], MARIO ALDAPE-PÉREZ[1], AND ALEJANDRO RODRÍGUEZ-MOLINA[2], (Member, IEEE)

[1]Postgraduate Department, Instituto Politécnico Nacional, CIDETEC, Mexico City 07700, Mexico
[2]Tecnológico Nacional de México/IT de Tlalnepantla, Research and Postgraduate Division, Estado de México 54070, Mexico

Corresponding author: Javier Maldonado-Romo (jmaldonador0501@alumno.ipn.mx)

This work was supported in part by the Secretaría de Investigación y Posgrado (SIP), and in part by the Comisión de Operación y Fomento de Actividades Académicas (COFAA) of the Instituto Politécnico Nacional (IPN).

ABSTRACT Micro aerial vehicles (MAVs) can make explorations in 3D environments using technologies capable of perceiving the environment to map and estimate the location of objects that could cause collisions, such as Simultaneous Localization and Mapping (SLAM). Nevertheless, the agent needs to move during the environment mapping, reducing the flying time to employ additional activities. It has to be noted that adding more devices (sensors) to MAVs implies more power consumption. Since more energy to perform tasks is required, growing the dimensions of MAVs limits the flying time. Contrarily, Generative Adversarial Networks (GAN) have demonstrated the usefulness of creating images from one domain to another, but the GAN domain changes require a large number of samples. Therefore, an interoperability coefficient is employed to determine a minimum number of samples to connect the different domains. In order to prove the coefficient, the performance to estimate the depth and semantic mask between authentic and virtual samples with the number limited of samples is analyzed. Consequently, an RGB-D sensor can be replaced by a few samples of a real scenario based on GANs. Although GAN allows creating images with depth and semantic mask information, there is an additional problem to be tackled: the presence of intrinsic noise, where a simple GAN architecture is not enough. In this proposal, the performance of this solution against a physical RGB-D sensor (Microsoft Kinect V1) and other state-of-the-art approaches is compared. Experimental results allow us to affirm that this proposal is a viable option to replace a physical RGB-D sensor with limited information.

INDEX TERMS Computer vision, perception environment, 3D mapping, machine learning.

I. INTRODUCTION

Robotics is a research area whose fundamental challenges have been obstacle detection and collision avoidance. Therefore one of the major topics to be investigated in this field is environment perception [1]. A common way to perceive is through sensory devices that translate stimuli received from the environment into interpretable data [2], where cameras and Inertial Measurement Units (IMU) are used as a complement for Global Navigation Satellite Systems (GNSS) in outside environments [3]. Typically, IMU is designed considering two main modules: accelerometer and gyroscope. When combined into a single device, they can create an array of information. However, the incorporation of arrays of sensors and embedded devices in Unmanned aerial vehicles (UAVs) is limited by the power consumption of all devices and vehicle dimensions. An emerging area of mechatronic design is applying artificial intelligence techniques to optimize the design of UAVs, considering both aspects [4]. Visual perception allows knowing the features of the environment [5], [6], while spatial context adds intelligence to autonomous systems to interpret the environment in order to develop a path planning solution [7].

In mobile robotics, there are three challenges to face when navigating in closed areas, especially when considering three dimensions (3D navigation) [8], [9]. The first is collision

The associate editor coordinating the review of this manuscript and approving it for publication was Seifedine Kadry.

detection and obstacle avoidance, the second consists of dynamic route planning (finding an alternative route when an obstacle is detected), and the third is the energy consumption of the entire system [10]–[12]. According to [13], the path planning problem has two approaches to describe it: the first one, called direct form, considers the robotic system as a point of reference, while the second, called indirect form, focus on the navigation environment. In the first approach, navigation environment analysis is about movement generated by the system, and we examine it from a first-person perspective. On the other hand, in indirect form, the robotic system is considered a particle; therefore, we analyze the environment from a third-person perspective.

Recently, technologies have been developed based on specific application devices that facilitate navigation indoors. RGB-D sensors are used together with RGB data for objects tracking [14], [15], as well as to improve objects classification accuracy [16]. Inertial Measurement Units (IMU) are typically used to improve location, avoid collisions [17], [18], as well as to map the environment based on vehicle movement [19], [20]. Other alternatives such as Lidar sensors [21], [22] and ultrasonic sensors [23], [24], both can perform the same tasks properly.

Modern exploration systems need to be efficiently designed and equipped with advanced perception systems to complete the task successfully. This reason implies a compromise between energy consumption and exploration autonomy. One of the main limitations MAVs have to explore 3D spaces is the duration of the batteries available in the vehicle. There is a direct relationship between the number of devices used in the environment perception system and their energy. When the more devices in the system, the greater the energy consumption. There is also an inverse relationship between energy consumption and flight autonomy. The higher the energy consumption, the shorter the battery life, and therefore the exploration time is less.

Hence the importance of optimizing the resources available to the MAV. Most of these vehicles already have a built-in camera, so this resource can be taken advantage of and used as a perception system to estimate authentic images' depth and semantic mask without adding additional devices.

This paper proposes a double-GAN-based architecture with noise reduction to estimate authentic images' depth and semantic mask using information generated by a virtual environment representation dataset with limited samples. This approach can effectively represent an RGB-D sensor using a few samples of a real scenario based on a double-GAN approach.

The remainder of the paper is organized as follows. Section II reviews and discusses previous works related to external environment perception systems. We describe in Section III the GAN architecture. Likewise, in Section IV, we introduce the interoperability coefficient to determinate the limited samples and the architecture with noise reduction to estimate the depth and semantic mask of real images. Experimental settings description are presented in Section V. Results analysis is presented in Section VI. Finally, some conclusions are presented in Section VII.

II. RELATED WORKS

It is well known that Machine Learning (ML) allows improving perception analysis in the robotics field [25]. However, there is less evidence of RGB-D sensor development using few samples of a real scenario by ML techniques. For example, during the last decade, perceiving the environment through RGB-D sensors has regained interest. Likewise, physical sensors such as Microsoft Kinect V1 [26] have been widely used to develop multiple solutions in the field of perception, providing autonomy to ground-vehicles [27], [28]. This RGB-D sensor offers adequate performance to perceive the environment [29] and detect obstacles [30]–[32] improving navigation [33]. Nonetheless, Microsoft Kinect V1 has specific features that limit its use in small spaces. This limitation is in terms of perception range, at least 40 cm. Consequently, objects must be beyond this distance to be perceived with a coverage range of up to 4 meters. Furthermore, Microsoft Kinect V1 also has a coverage range configuration of up to 6 meters, with an 80 cm offset. Thus, it is unsuitable to use this sensor in small scenarios [34].

Computer vision processing has been used to analyze and find features on input data [35], [36]. In the same way, ML algorithms have been widely used in classification [37], [38] and regression tasks [39], [40]. As a result of the intersection of both paradigms, another research area, called Generative modeling, has emerged. Generative modeling uses Generative Adversarial Networks (GANs) to generate realistic examples across various problem domains. This type of network automatically learns the regularities in input data so that the model can be used to generate new examples that plausibly could have been drawn from the original dataset.

GANs are based on a competition approach between two types of neuronal networks: generative and discriminative network [41]. The first one is responsible for generating data from a noisy source while the discriminative network is in charge of extracting a set of known characteristics of examples to validate the generator model [42]. The generative network generates candidates while the discriminative network evaluates them.

Besides, GANs have also been used for image transformation to map data into a different domain [43] and generate data to create an image with different machine learning approaches [44]–[46]. This topic is essential in our approach because we can obtain data from a conventional camera and process them to generate either a depth image, semantic mask image, or both using GANs. The depth data describe a point in a 3D space, whereas the semantic mask represents the information that composes an object.

For this reason, in this work, it is proposed to generate a representation of an RGB-D sensor using information from samples created by a virtual environment to estimate the depth and semantic mask of authentic images to map a 3D environment.

In this way, it is taken advantage of the camera available on most MAVs to obtain an efficient depth sensor without adding additional devices that affect the aerodynamics and structure of the vehicle and energy consumption and consequently flight autonomy.

III. GAN ARCHITECTURE

GANs is an architecture that uses two neural networks to generate new synthetic data instances that plausibly could be considered real data. As it is shown in Figure 1, a simple GAN architecture is composed of an auto-encoder as the generator network and a deep convolutional network as the discriminator, where specific parameters are as follows: kernel size is equal to 3, the stride is equal to two in the generator network, and stride is equal to one in the discriminator network, Relu as activation function in both networks, and Adam as optimizer. Figure 2 shows the architecture parameters for generator and discriminator, respectively.

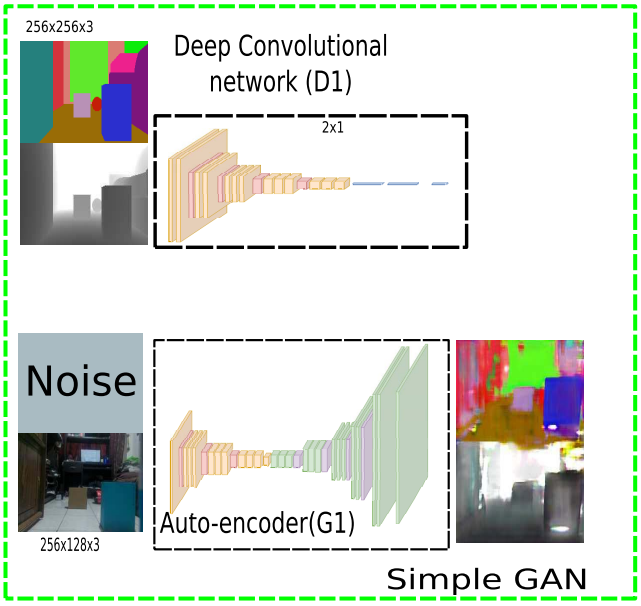


FIGURE 1. Simple GAN architecture.

In order to estimate the depth and semantic mask of authentic images using information generated by a virtual environment representation, performance provided by a simple network is insufficient because each set of samples must share some in-domain and cross-domain features [47], [48]. For this reason, it is necessary to have an intermediary element between domains to minimize noise and share some cross-domain features.

A. GAN COST FUNCTIONS

The system requires an adequate implementation of a GAN that allows generating a representation of an RGB-D sensor using information from samples created by a virtual environment to estimate the depth and semantic mask of authentic images to map a 3D environment. It is worth mentioning that GANs contain regularization terms that allow for adequate training. This set of rules is called the cost function. In order to get better performance, an optimization process needs to be incorporated. Specifically, this process involves the maximization of the generator network cost function $G_{1,2}$, the minimization of the discriminator cost function $D_{1,2}$, and the minimization of the noise source cost function $Z_{1,2}$. The cost functions of a GAN network are derived from the calculus of entropy [49]. The principal characteristics of the GAN are given by the 1, 2 and 3 definitions.

Definition 1: Let n be the number of samples, let $D_{1,2}$ be the cost function of the discriminator network, let $G_{1,2}$ be the cost function of the generator network, and let $Z_{1,2}$ be the noise source. Maximization of the cost function of the discriminator network is obtained according to the following expression:

$$M_{cf_{D_{1,2}}} = \frac{1}{n} \cdot \sum_{i=0}^{n} log(D_{1,2}(i)) + log(1 - D_{1,2}(G_{1,2}(Z_{1,2}^{i}))) \tag{1}$$

Definition 2: Let n be the number of samples, let $D_{1,2}$ be the cost function of the discriminator network, let $G_{1,2}$ be the cost function of the generator network, and let $Z_{1,2}$ be the noise source. Minimization of the cost function of the generator network is obtained according to the following expression:

$$m_{cf_{G_{1,2}}} = \frac{1}{n} \cdot \sum_{i=0}^{n} -log(D_{1,2}(G_{1,2}(Z^{i}))) \tag{2}$$

Definition 3: Let $D_{1,2}$ be the cost function of the discriminator network, let $G_{1,2}$ be the cost function of the generator network and let $Z_{1,2}$ be the noise source. The full cost function of a simple GAN architecture is obtained according to the following expression:

$$GAN_{cf_{1,2}} = M_{cf_{D_{1,2}}} + m_{cf_{G_{1,2}}} \tag{3}$$

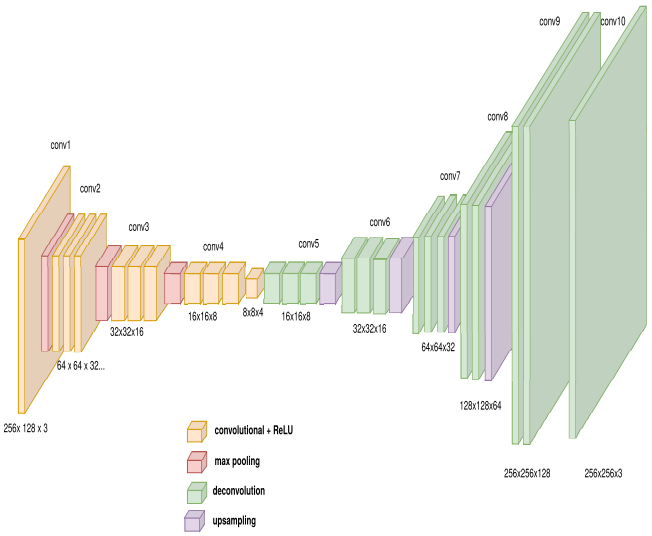
IV. PROPOSED WORK

In this section, the generation of the dataset is described. In addition, a coefficient is employed to determine a limited number of samples of a physical scenario. Finally, the proposed architecture is illustrated to generate a virtual sensor with limited samples.

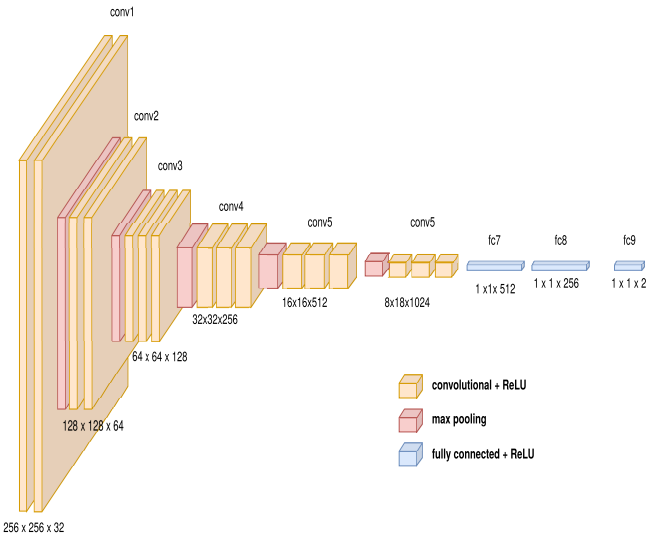
A. 3D VIRTUAL ENVIRONMENT AND SAMPLES

Considering that access to the physical environment is limited, a 3D virtual environment is built based on a limited number of samples of the environment. Virtual samples generated have a resolution of 256 × 144, with an aspect ratio of 16:9. Thus, each pixel represents a centimeter at a distance of 1.5 m with a Field of View (FOV) of 82.6, as it is shown in Figure 3.

AirSim framework generates depth images in a range of 100 m. For this research, the framework was modified to create depth images in a range of 2 m and 5 m. For example, Figure 4 shows depth images in a range of 2 m in which the size for each sample is 256 × 128 with three channels.



(a) Model features for the GAN's generator, which estimates the virtual samples from real inputs.



(b) Model discriminator's features to validate the training in the GAN.

FIGURE 2. Simple GAN features.

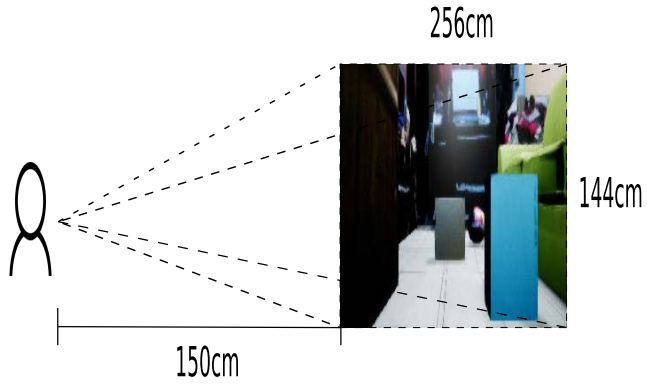


FIGURE 3. Resolution of the virtual environment.

Likewise, the dataset is composed of three domains, Figure 4a represents the domains of authentic samples, Figure 4b shows the virtual representation domain, and Figure 4c displays the depth and mask semantic domain.

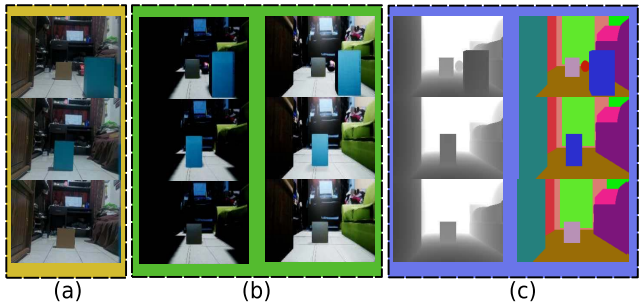


FIGURE 4. Kind of samples for each domain. (a) Physical world. (b) Virtual representation. (c) Semantic and depth information.

B. SIMILARITY BETWEEN IMAGES

One of the tools for assessing the correlation between two images is the Histogram of Oriented Gradients (HOG) [51]. This algorithm allows measuring the comparison between real and virtual representation. This descriptor obtains a characteristic vector for each of the samples and computes a coefficient that indicates the similarity level, whose hyperparameters are: orientation equal to 8, pixels per cell equal to 32×32, and cells per block equal to 4×4. For example, Figure 5 shows a physical sample and its virtual representation with two different detail levels. The first variation has essential lighting, and the second has a more significant number of directional lighting sources and materials that give more realism to the virtual environment.

FIGURE 5. Histogram of oriented gradients for each sample. (a) Physical world. (b) Virtual representation with an essential light source. (c) Virtual samples with lights and materials.

Table 1 shows correlation measurements between 30 physical world samples and their virtual representation with two different detail levels. The correlation coefficient of more detailed samples (lights and materials) is higher than

essential light source samples since lights increase detail level. However, the correlation coefficient between virtual samples created with video game engines and real examples is not high enough to claim that an adequate representation of the physical world is obtained.

TABLE 1. Correlation between physical world and virtual samples.

	Simple environment Virtual-Real	Environment with lights-materials Virtual-Real
Factor Correlation (mean)	0.3708	0.5490
Factor Correlation (std)	0.0824	0.0755

C. INTEROPERABILITY COEFFICIENT FOR CONNECT VIRTUAL AND REAL ENVIRONMENTS

In [52] an interoperability coefficient was introduced, which consists of a minimum number of physical samples to connect the virtual and physical domains using the GAN characteristics. This coefficient is composed of a correlation factor generated by HOG and the GAN's entropy, and it is expressed in 4.

$$C_{inter} = \frac{\sum_{k}^{N_{real}} HOG(x_{r_k}, x_{v_k})}{N_{real}} \cdot \left(1 - \frac{\sum_{i}^{N_{GAN}} \frac{-\sum_{y_i}\sum_{y_i'} P_{Y_iY_i'}(y_i,y_i') log P_{Y_iY_i'}(y_i,y_i')}{H(y_i)+H(y_i')}}{N_{GAN}}\right) \tag{4}$$

In this proposal, the HOG correlation is 0.5490, and the interoperability coefficient is 0.5047 in 43 physical world samples shown in Table 2. For this reason, it is recommended to take the number of samples when the interoperability coefficient is upper than 0.50. Thus, the details in the virtual representation are lower than the authentic sample. We consider that the virtual representation must have enough information that allows deep learning to use textures. Furthermore, it is observed that if the number of virtual representation samples increases their details in light and material, then the interoperability coefficient must increase, and the number of samples can be less. When the joint entropy is low, the data dispersion is similar between the GAN architecture and virtual representation samples. However, we are in a case where the authentic information is limited, and the samples do not have enough details. In this way, it is avoided taking more samples.

TABLE 2. Minimum real samples estimation for building VR environment.

Number of samples	Join entropy	Interoperability coefficient
10	0.5894	0.1985
20	0.2567	0.2935
30	0.1564	0.4465
43	0.0957	0.5047

D. ARCHITECTURE

Several methodologies have been proposed to solve navigation problems, but most of these approaches require physical world samples for path planning and MAV training. This behavior implies that used samples to train must come from the same domain. Therefore, the MAV training has to realize in an authentic environment. Besides, this type of training requires high consumption of time, costs of operating, and maintenance on the MAV.

In this way, a double-GAN-based architecture with noise reduction is proposed to estimate authentic images with depth and semantic mask using virtual samples. The first GAN creates a virtual sample of the authentic environment. This architecture is an intermediary element between domains (virtual and real) used to minimize noise and share some cross-domain features. Figure 6 shows the parameters of the Double GAN architecture. An additional layer is added in generator networks because the output image size is 256×256 pixels.

Therefore, the samples compose three different domains, namely: the physical world, virtual environment, and the third domain consisting of GAN generated images representing semantic mask and depth information of physical world samples, as is shown in Figure 7. Likewise, we propose a three-module architecture, as it is shown in Figure 8. The first module is an intermediary element between real and virtual domains. This module converts an authentic image to a virtual representation throughout the GAN architecture. On the other hand, the second module generates the depth and semantic mask samples. Finally, Definition 4 describes the third module that minimizes the noise.

Definition 4: Let n be the number of samples, let y_i' be a generated image, and let y_i be an original image. The minimization cost function for noise reduction is obtained according to the following expression:

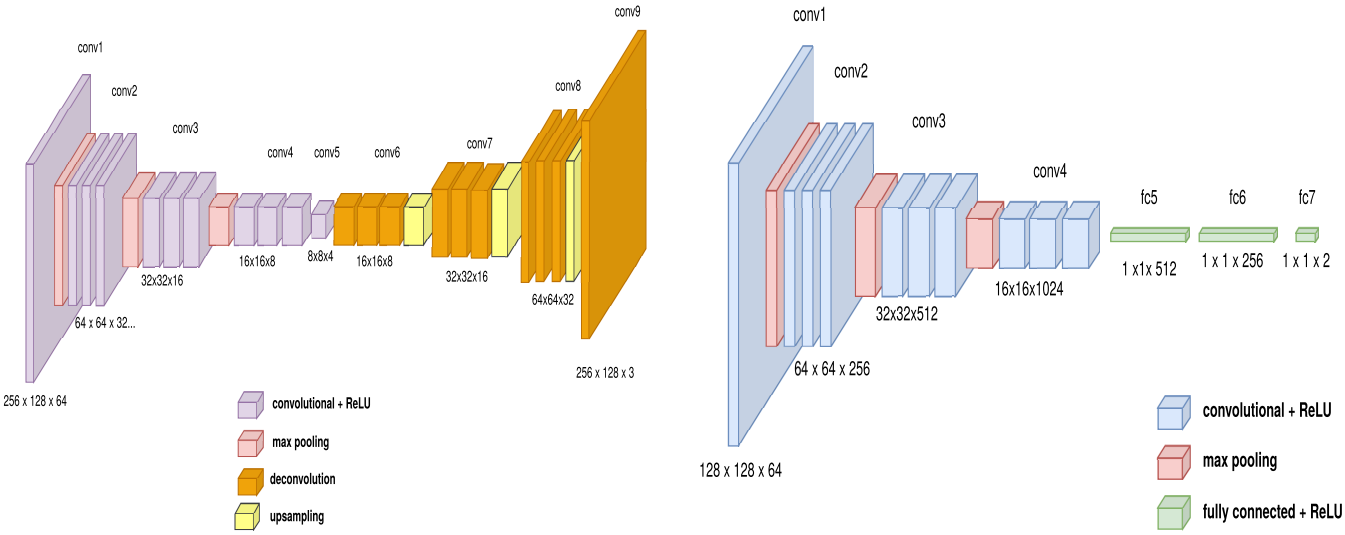
$$m_{cf_{NR}} = \frac{1}{n} \cdot \sum_{i=0}^{n} |y_i' - y_i| \tag{5}$$

This architecture provides data stability between three domains and maintains some cross-domain features to represent the depth and semantic mask. Furthermore, it is possible to create different virtual environments with the proposed architecture to generate many virtual samples with limited information of an authentic environment.

V. EXPERIMENTAL PHASE

The proposed architecture, shown in Figure 8, was implemented in a g4dn.xlarge instance in Amazon Web Services (AWS) with the following specifications: 4 VCPU XEON 8259CL 2.5GHz, 16GB RAM, 125GB SDD storage, with NVIDIA Tesla T4 GPU with 320 Tensor Core with 16GB RAM. Algorithm 1 was implemented in Tensorflow 2.3. Training time was 3 hours and 10 minutes.

Along with the experimental phase, 43 samples were used for the first GAN architecture (Module 1); for the second module, 1000 samples were generated from the virtual environment; finally, the same 1000 virtual samples were used for the third module. These virtual samples were processed



(a) Model features for the first domain change generator.

(b) Model features for the first domain change discriminator.

FIGURE 6. First stage features for the double GAN.

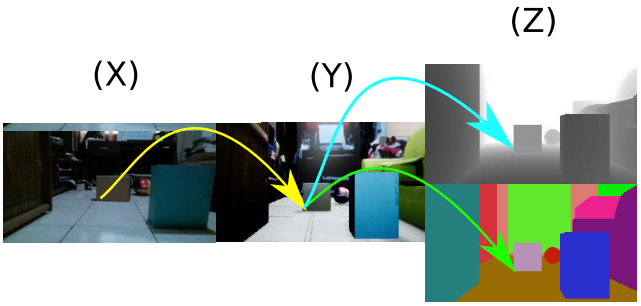


FIGURE 7. Domain change problem.

through a serial convolutional network to reduce data randomness and maintain better virtual sample uniformity.

The physical environment is composed of three different types of objects. Two of these objects are cardboard boxes of different colors and sizes, and the third is a ball. Microsoft Kinect V1 minimum perception range is 40 cm in a 4m range configuration. Therefore objects must be beyond 40 cm to be perceived with a coverage range of up to 4 meters. In this way, we have flat surfaces and curved surfaces to measure consistent data. The depth in virtual samples was estimated in two ranges: 2 meters and 5 meters. For the 2 m samples, each pixel is equal to 0.7843 cm, while for 5 m samples is 1.9607 cm.

The stability of the complete cost function of both architectures (Simple GAN and Double GAN), after 1000 epochs, is shown in Figure 9. The graph shows that GAN_1 network is less stable than GAN_2 network. On the other hand, it can also be observed that GAN_2 network improves data output stability. The performance of each module of both architectures is shown in Figure 10. The graph shows that D_2 and G_2 networks have more stable behavior than D_1 and G_1 networks. Likewise, the noise reduction module performance is notable.

Algorithm 1 Algorithm for Estimating the Depth and Semantic Mask

Input: a set of 43 images with physical domain.
Input: a set of 300 images with virtual domains.
Output: a set of estimated depth for each sample.
Initialization:
1: Load dataset.
2: Apply augmented data to increase samples variety.
3: Create Generative network model.
4: Create Discriminative network model.
5: Create Noise reduction model.
6: Define loss function for generative and discriminative networks by mean value.
7: Define loss function for noise reduction model by MSE value.
Loop training
8: **for** $i = 0$ to 300 **do**
9: Run sample batch on generative and discriminative networks to change to first domain.
10: Run sample batch on generative and discriminative networks to change to second domain.
11: Run noise reduction model.
12: Update gradients.
13: **end for**
14: **return** estimated depth samples

A. METRICS

Evaluation criteria are based on error and accuracy metrics proposed by Eigen *et al.* [53] to evaluate and compare the performance of depth estimation methods. These metrics are

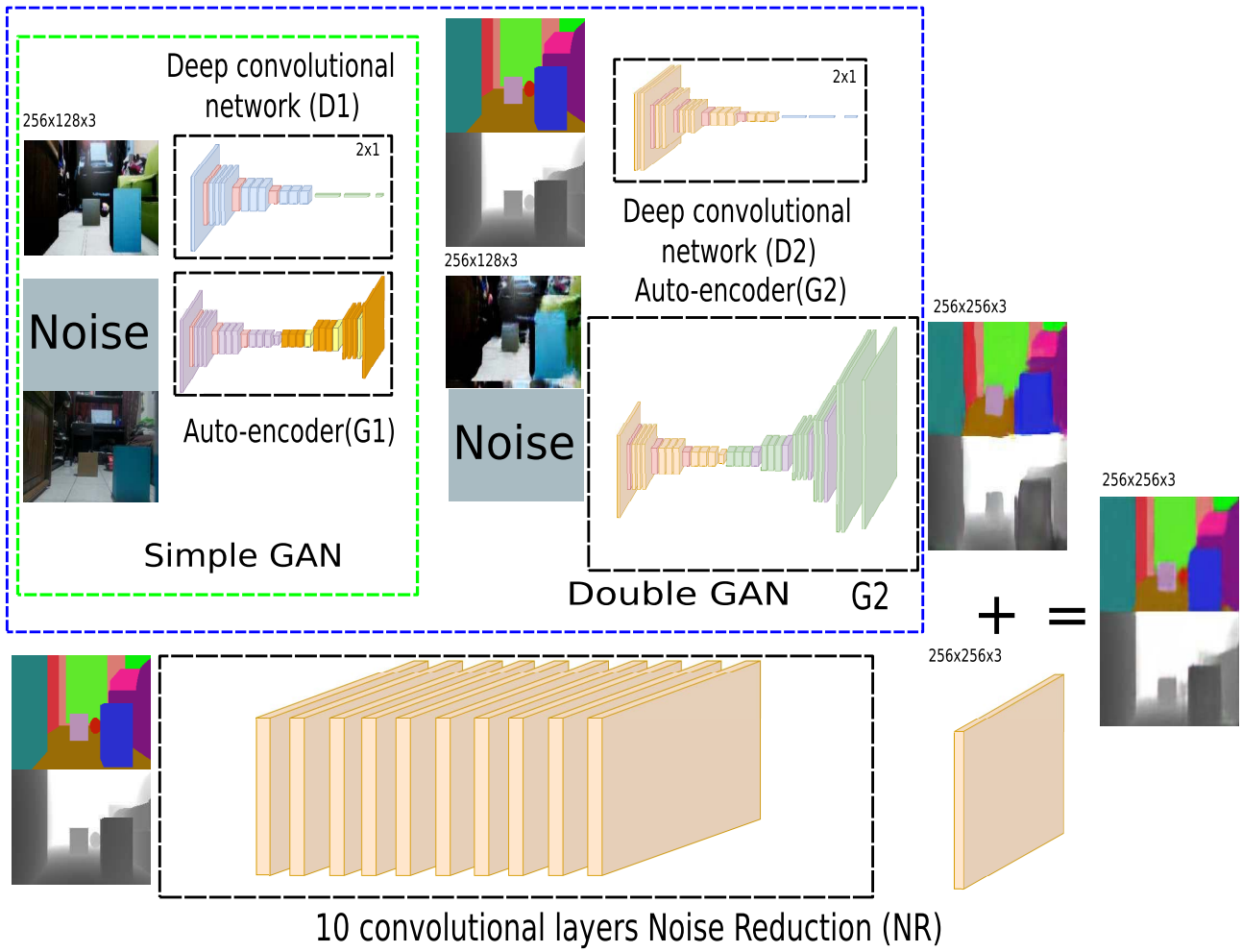


FIGURE 8. Architecture to create depth and semantic mask samples generated in the virtual environment from physical world.

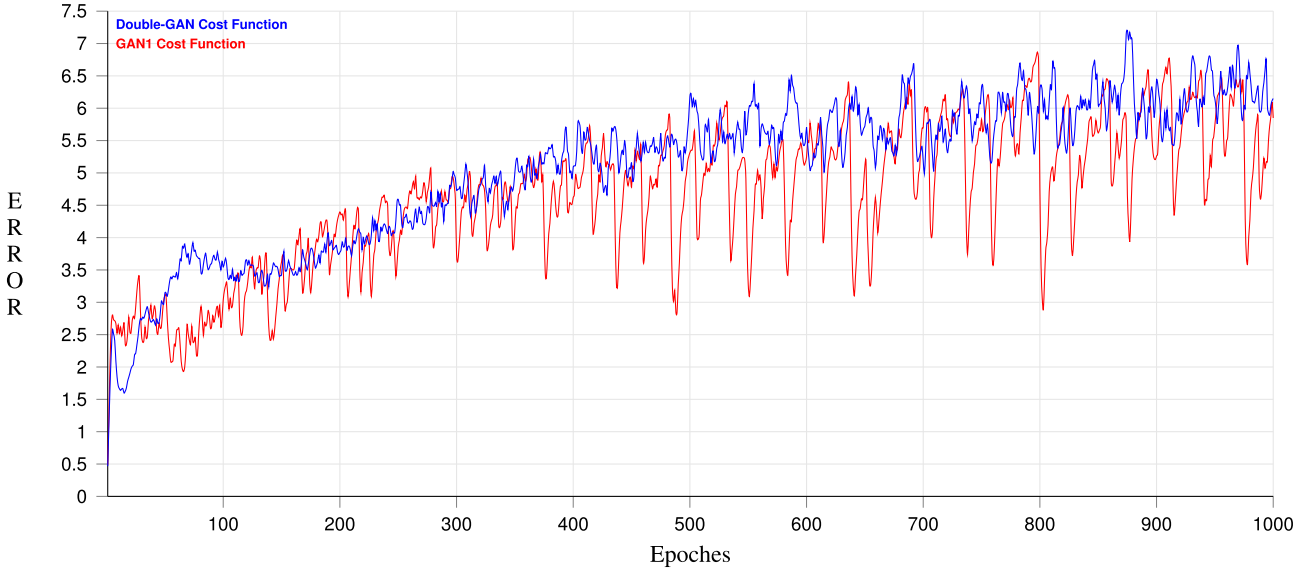


FIGURE 9. Behavior of complete cost function of both networks (GAN_1 and GAN_2).

formulated as follows, where Y_p is a pixel in-depth image Y (ground truth image), $\hat{Y}_p$ is a pixel in the estimated depth image $\hat{Y}$, and k is the total number of pixels for each depth image.

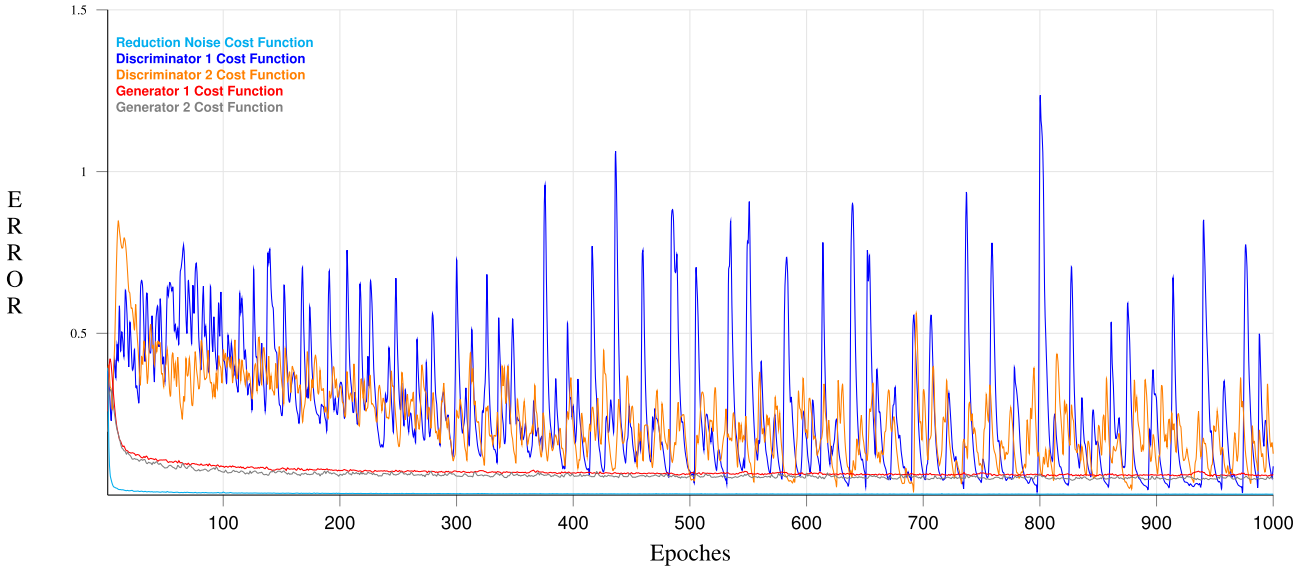


FIGURE 10. Behavior of cost function components of both networks (GAN_1 and GAN_2).

Definition 5: Relative error (rel) is obtained according to the following expression:

$$\text{rel}(Y_p, \hat{Y}_p) = \frac{1}{k}\sum_{p=1}^{k}\frac{\left|Y_p - \hat{Y}_p\right|}{Y_p} \tag{6}$$

Definition 6: Average (log_{10}) error is obtained according to the following expression:

$$log_{10}\ \text{error}(Y_p, \hat{Y}_p) = \frac{1}{k}\sum_{p=1}^{k}\left|log_{10}(Y_p) - log_{10}(\hat{Y}_p)\right| \tag{7}$$

Definition 7: Root mean-squared error (RMSE) is obtained according to the following expression:

$$\text{RMSE}(Y_p, \hat{Y}_p) = \sqrt{\frac{1}{k}\sum_{p=1}^{k}(Y_p - \hat{Y}_p)^2} \tag{8}$$

Definition 8: Accuracy with threshold (t): Percentage of Y_p s.t. $\max(\frac{Y_p}{\hat{Y}_p}, \frac{\hat{Y}_p}{Y_p}) = \delta < t, t \in [1.25, 1.25^2, 1.25^3]$

VI. RESULTS

The performance of this proposal was evaluated using metrics presented in Section V-A. These metrics have been widely used for depth estimation with adversarial training [54]–[60]. Table 3 shows performance results achieved by this proposal in three different stages: Simple GAN, Double-GAN, and Double-GAN with noise reduction (Double-GAN-NR).

Table 4 shows depth estimation average values and standard deviation of 50 samples of flat surfaces on three different objects. The DG-2 and DG-5 results correspond to the architecture output without noise reduction at two distances: 2 m and 5 m (Figure 8). Therefore, Double-GAN with noise reduction results is labeled as Double-GAN-NR-2 and Double-GAN-NR-5 for both distances.

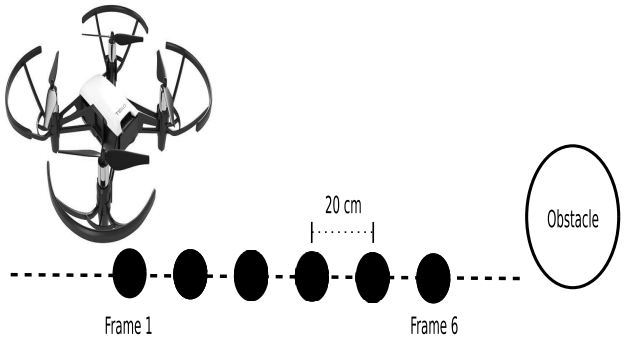


FIGURE 11. Description of the sample capture procedure.

As it is shown in Table 4, DG-2 and DG-5 present less promising results since they achieve depth estimation results that are further from the ground truth value. Microsoft Kinect shows better results than DG-2 and DG-5 models. However, depth estimation results achieved by this sensor exhibit an inverse relationship between distance to be detected and measurement precision. That is, as distance range increases, depth estimation precision decreases. Regarding noise reduction models, we can say that Double-GAN-NR-2 shows better performance than Double-GAN-NR-5. This behavior is because the double-GAN-NR-2 model achieves depth measurements closer to the ground truth value. Table 5 shows how each of the models obtains close distance measurements concerning the ground-truth value. As can be seen, Double-GAN-NR-2 shows the best performance.

Likewise, Figure 11 shows a graphical description of the sample capture procedure to build a 3D environment,

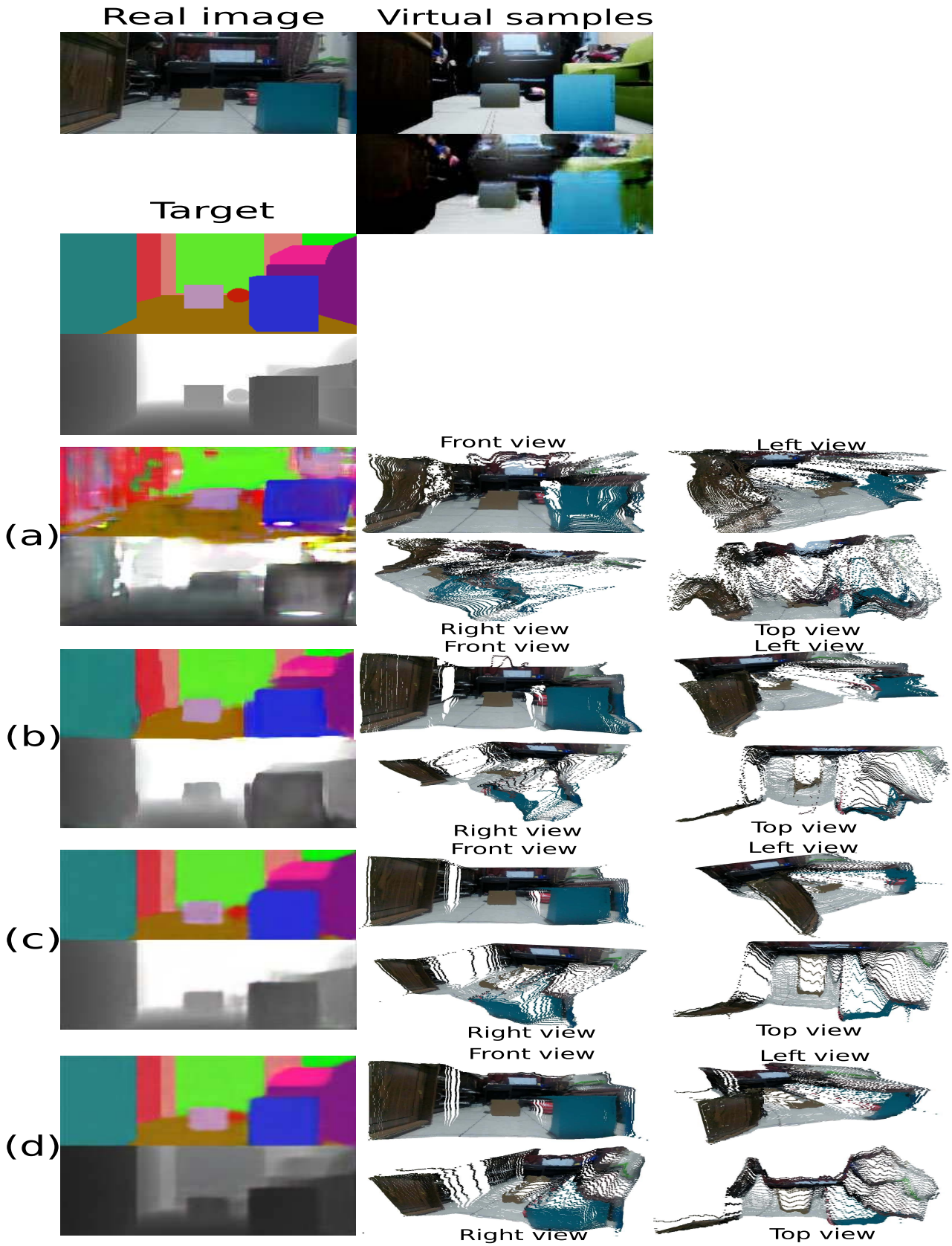


FIGURE 12. Results of the experiment. (a) Simple-GAN for 2m. (b) DG-2m. (c) Double-GAN-NR-2m. (d) Double-GAN-NR-5m.

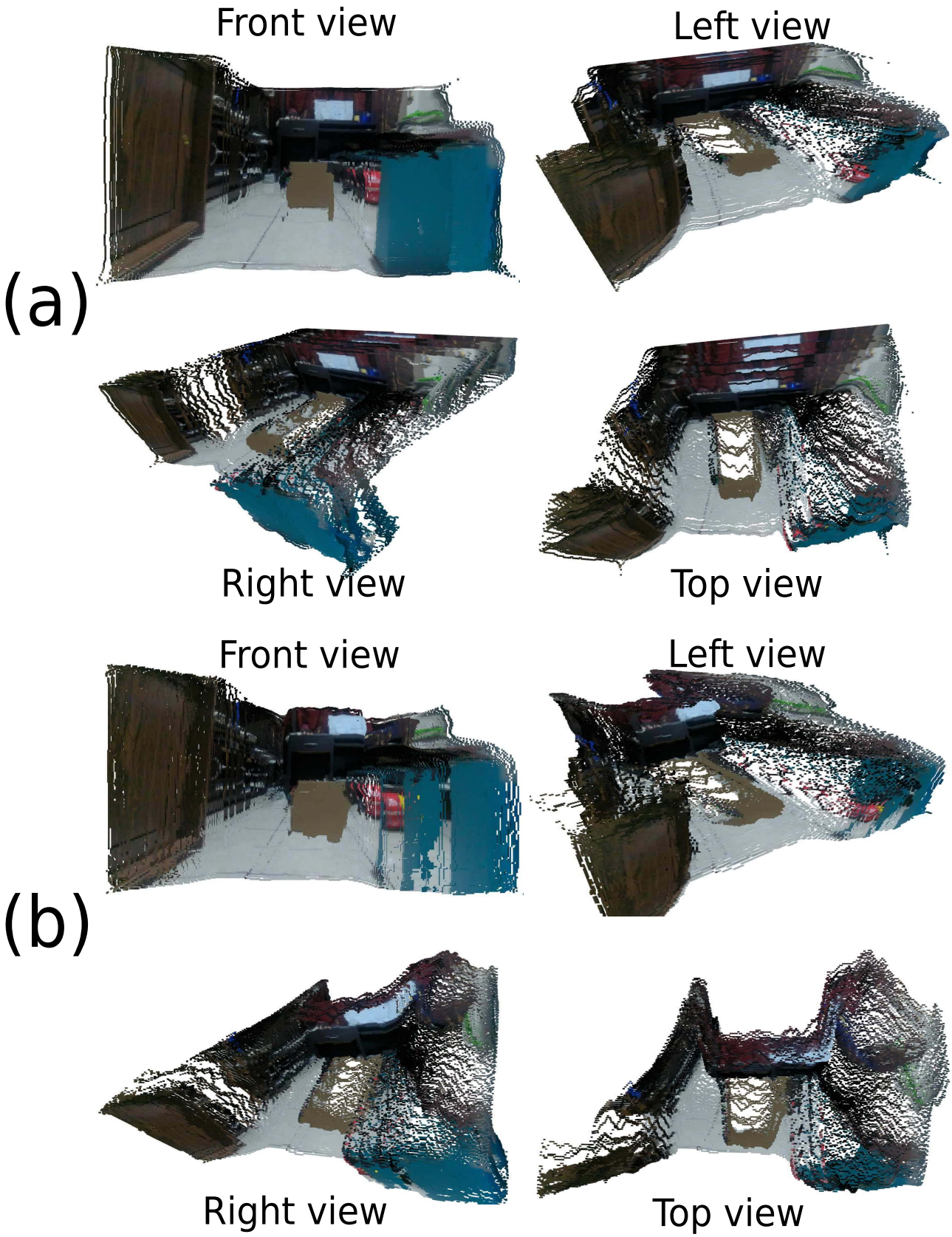


FIGURE 13. A 3D environment built with six frames using (a) 2 m. and (b) 5 m. depth configurations.

TABLE 3. Performance of this proposal and its standard deviation in 50 samples. Where ↓ Low is better, and ↑ Up is better.

Model	rel - std ↓	rms - std ↓	log_{10} - std ↓	δ_1 - std ↑	δ_2 - std ↑	δ_3 - std ↑
Simple-GAN	0.428 - 0.028	0.526 - 0.032	0.128 - 0.031	0.729 - 0.034	0.827 - 0.049	0.862 - 0.046
Double-GAN	0.208 - 0.030	0.473 - 0.038	0.116 - 0.016	0.735 - 0.096	0.868 - 0.082	0.947 - 0.041
Double-GAN-NR	**0.189 - 0.011**	**0.433 - 0.009**	**0.108 - 0.015**	**0.792 - 0.020**	**0.897 - 0.016**	**0.976 - 0.021**

TABLE 4. Experimental results: average value and standard deviation of depth evaluation on depth images of 2 m and 5 m using Double GAN (DG-2 & DG-5), Double GAN with noise reduction (Double-GAN-NR-2 & Double-GAN-NR-5), and Microsoft Kinect.

Dis. (cm)	DG-2		DGNR-2		DG- 5		DGNR-5		Kinect	
	mean	std	mean	std	mean	std	mean	std	mean	std
50	62.754	11.770	50.896	2.110	96.274	11.519	62.468	2.280	52.040	1.459
70	82.417	6.430	69.693	2.453	106.038	8.746	83.925	3.747	66.581	1.857
90	108.509	6.934	88.262	2.366	125.596	8.077	110.826	3.030	85.099	1.293
110	125.810	4.106	106.479	1.399	138.312	7.268	126.036	3.481	103.892	1.372
130	144.350	4.605	125.142	4.009	148.225	8.173	142.672	3.660	122.214	1.336
150	162.394	4.950	142.627	3.558	164.658	8.384	159.190	3.773	141.501	1.558

TABLE 5. Experimental results: absolute difference obtained by each model with respect to the ground truth value.

Dis. (cm)	DG-2	Double-GAN-NR-2	DG-5	Double-GAN-NR-5	Kinect
50	12.754	**0.896**	46.275	12.468	2.040
70	12.418	**0.306**	36.039	13.926	3.419
90	18.510	**1.737**	35.597	20.826	4.901
110	15.811	**3.521**	28.313	16.036	6.107
130	14.350	**4.858**	18.226	12.672	7.786
150	12.395	**7.372**	14.658	9.191	8.499

as suggested in [61]. Figure 12 shows 3D environment representation obtained with compared GAN methods. Finally, Figure 13 shows Microsoft Kinect results for 3D scene representation with 2 meters and 5 meters depth configurations.

In order to validate the Double-GAN approach, we compared the complete set of virtual samples with original images. Figure 12a shows randomness behavior that appears on flat surfaces. Therefore, the intermediary module minimizes noise and shares cross-domain features between domains. As can be seen in Figure 12b, more consistent results are obtained. However, when performing 3D representation, irregularities are observed in flat sections of the objects. Figure 12c and Figure 12d show Double-GAN architecture with noise reduction results. Depth in virtual samples was estimated in two ranges: 2 meters and 5 meters, respectively. As can be seen in Figure 12d, the best depth estimation results were obtained with the 5m configuration.

Results of Tables 4 and 5 show that the Double-GAN-NR-2 model can estimate image depth successfully in a range from 50 cm to 150 cm, considering a maximum offset of 7.4 cm. On the other hand, the maximum offset required by the Double-GAN-NR-5 model to estimate the depth of an image in the same range as the Double-GAN-NR-2 model is 20.9 cm, while for the Microsoft Kinect Sensor is 8.5 cm.

A moderately detailed 3D image, but with an adequate lighting level, allows creating a virtual representation of the authentic scene without having access to the authentic environment. Experimental results are shown in Figure 12, and Figure 13 confirm that it is possible to obtain a real scene representation using depth and semantic mask estimation in virtual samples using a Double-GAN approach with a limited number of samples.

VII. CONCLUSION

Due to the interoperability coefficient providing a limited number of samples of authentic environments, we can employ a virtual environment to reduce the samples of the environment with difficult access.

Likewise, experimental results confirm that a low-cost depth sensor developed through virtual samples of an authentic environment can be obtained using GANs. Besides, the results suggest that a simple GAN approach for semantic mask and depth estimation in virtual samples of an authentic environment is not enough to achieve competitive performance. Conversely, a Double-GAN approach can estimate depth in virtual samples of an authentic environment showing competitive performance.

Furthermore, experimental results confirm that the Double-GAN approach provides much information on a conventional camera to estimate semantic mask and depth in a 3D environment that requires continuous depth estimation with limited samples. Thus, we avoid adding more devices for keeping the time of flight on MAVs in a controlled indoor environment where the illumination is constant.

As future work, we proposed to use the Double-GAN approach to detect moving obstacles in MAVs' paths to implement obstacle avoidance algorithms and optimal navigation path planning.

REFERENCES

[1] M. H. Ang, O. Khatib, and B. Siciliano, *Encyclopedia of Robotics*. Berlin, Germany: Springer, 2020, doi: 10.1007/978-3-642-41610-1.

[2] K. Kalantar-Zadeh, *Sensors*. Cham, Switzerland: Springer, 2013, doi: 10.1007/978-1-4614-5052-8.

[3] J. Collin, "MEMS IMU carouseling for ground vehicles," *IEEE Trans. Veh. Technol.*, vol. 64, no. 6, pp. 2242–2251, Jun. 2015, doi: 10.1109/TVT.2014.2345847.

[4] H. Zhu, H. Nie, L. Zhang, X. Wei, and M. Zhang, "Design and assessment of octocopter drones with improved aerodynamic efficiency and performance," *Aerosp. Sci. Technol.*, vol. 106, Jun. 2020, Art. no. 106206, doi: 10.1016/j.ast.2020.106206.

[5] Y. Tian, X. Chen, H. Xiong, H. Li, L. Dai, J. Chen, J. Xing, J. Chen, X. Wu, W. Hu, Y. Hu, T. Huang, and W. Gao, "Towards human-like and transhuman perception in AI 2.0: A review," *Frontiers Inf. Technol. Electron. Eng.*, vol. 18, no. 1, pp. 58–67, 2017, doi: 10.1631/fitee.1601804.

[6] O. Ciftcioglu, M. S. Bittermann, and I. S. Sariyildiz, "Towards computer-based perception by modeling visual perception: A probabilistic theory," in *Proc. IEEE Int. Conf. Syst., Man Cybern.*, Taipei, Japan, Oct. 2006, pp. 5152–5159, doi: 10.1109/ICSMC.2006.385126.

[7] G. Flores-Caballero and M. G. Villarreal-Cervantes, "Optimized path-planning in continuous spaces for unmanned aerial vehicles using meta-heuristics," *IEEE Access*, vol. 8, pp. 176774–176788, 2020, doi: doi:.10.1109/ACCESS.2020.3026666.

[8] M. Hassanalian and A. Abdelkefi, "Classifications, applications, and design challenges of drones: A review," *Progr. Aerosp. Sci.*, vol. 91, pp. 99–131, May 2017, doi: 10.1016/j.paerosci.2017.04.003.

[9] K. Dalamagkidis, "Classification of UAVs," in *Handbook of Unmanned Aerial Vehicles*, K. Valavanis and G. Vachtsevanos, Eds. Dordrecht, The Netherlands: Springer, 2015, doi: 10.1007/978-90-481-9707-1_94.

[10] J. Ding and Y. Fang, "Multi-strategy based exploration for 3D mapping in unknown environments using a mobile robot," in *Proc. Chin. Control Conf. (CCC)*, Guangzhou, China, Jul. 2019, pp. 4732–4738, doi: 10.23919/ChiCC.2019.8866080.

[11] C. Zhu, R. Ding, M. Lin, and Y. Wu, "A 3D frontier-based exploration tool for MAVs," in *Proc. IEEE 27th Int. Conf. Tools Artif. Intell. (ICTAI)*, Vietri Mare, Italy, Nov. 2015, pp. 348–352, doi: 10.1109/ICTAI.2015.60.

[12] G. Huang, Y. Lu, and Y. Nan, "A survey of numerical algorithms for trajectory optimization of flight vehicles," *Sci. China Technol. Sc.*, vol. 55, no. 9, pp. 2538–2560, Sep. 2012, doi: 10.1007/s11431-012-4946-y.

[13] S. Roland and N. Illah, *Introduction to Autonomous Mobile Robots*. London, U.K.: MIT Press, 1997.

[14] Z. Cai and L. Shao, "RGB-D data fusion in complex space," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Beijing, China, Sep. 2017, pp. 1965–1969, doi: 10.1109/ICIP.2017.8296625.

[15] F. Fang, K. Qian, B. Zhou, and X. Ma, "Real-time RGB-D based people detection and tracking system for mobile robots," in *Proc. IEEE Int. Conf. Mechatronics Autom. (ICMA)*, Takamatsu, Japan, Aug. 2017, pp. 1937–1941, doi: 10.1109/ICMA.2017.8016114.

[16] W. Khan, E. Phaisangittisagul, L. Ali, D. Gansawat, and I. Kumazawa, "Combining features for RGB-D object recognition," in *Proc. Int. Electr. Eng. Congr. (iEECON)*, Pattaya, Thailand, Mar. 2017, pp. 1–5, doi: 10.1109/IEECON.2017.8075877.

[17] S. T. Kebir, H. Kheddar, M. Maazouz, S. Mekaoui, A. Ferrah, and R. Mazari, "Smart robot navigation using RGB-D camera," in *Proc. Int. Conf. Appl. Smart Syst. (ICASS)*, Medea, Algeria, Nov. 2018, pp. 1–6, doi: 10.1109/ICASS.2018.8652025.

[18] S. Gatesichapakorn, J. Takamatsu, and M. Ruchanurucks, "ROS based autonomous mobile robot navigation using 2D LiDAR and RGB-D camera," in *Proc. 1st Int. Symp. Instrum., Control, Artif. Intell., Robot. (ICA-SYMP)*, Bangkok, Thailand, Jan. 2019, pp. 151–154, doi: 10.1109/ICA-SYMP.2019.8645984.

[19] X. Liu, B. Guo, and C. Meng, "A method of simultaneous location and mapping based on RGB-D cameras," in *Proc. 14th Int. Conf. Control, Autom., Robot. Vis. (ICARCV)*, Phuket, Thailand, Nov. 2016, pp. 1–5, doi: 10.1109/ICARCV.2016.7838786.

[20] U. Qayyum, Q. Ahsan, and Z. Mahmood, "IMU aided RGB-D SLAM," in *Proc. 14th Int. Bhurban Conf. Appl. Sci. Technol. (IBCAST)*, Islamabad, Pakistan, Jan. 2017, pp. 337–341, doi: 10.1109/IBCAST.2017.7868075.

[21] J. Li, X. He, and J. Li, "2D LiDAR and camera fusion in 3D modeling of indoor environment," in *Proc. Nat. Aerosp. Electron. Conf. (NAE-CON)*, Dayton, OH, USA, Jun. 2015, pp. 379–383, doi: 10.1109/NAE-CON.2015.7443100.

[22] H. Qin, J. Li, J. Wang, and Q. Wu, "Local map construction based on 3D-LiDAR and camera," in *Proc. 39th Chin. Control Conf. (CCC)*, Shenyang, China, Jul. 2020, pp. 3887–3891, doi: 10.23919/CCC50068.2020.9188499.

[23] C. C. Chang, C. Y. Chang, and Y. T. Cheng, "Distance measurement technology development at remotely teleoperated robotic manipulator system for underwater constructions," in *Proc. Int. Symp. Underwater Technol.*, Taipei, Japan, 2004, pp. 333–338, doi: 10.1109/UT.2004.1405598.

[24] A. Nishitani, Y. Nishida, and H. Mizoguch, "Omnidirectional ultrasonic location sensor," in *Proc. SENSORS*, Irvine, CA, USA, Oct. 2005, pp. 1–4, doi: 10.1109/ICSENS.2005.1597791.

[25] D. M. R. Fernandes and M. A. Ponti, *Machine Learning*. Cham, Switzerland: Springer, 2018, doi: 10.1007/978-3-319-94989-5.

[26] S. Kean, J. C. Hall, and P. Perry, "Behind the technology," in *Meet the Kinect*. New York, NY, USA: Apress, 2011, doi: 10.1007/978-1-4302-3889-8_2.

[27] N. A. Zainuddin, Y. M. Mustafah, Y. A. M. Shawgi, and N. K. A. M. Rashid, "Autonomous navigation of mobile robot using Kinect sensor," in *Proc. Int. Conf. Comput. Commun. Eng.*, Kuala Lumpur, Malaysia, Sep. 2014, pp. 28–31, doi: 10.1109/ICCCE.2014.21.

[28] O. Wasenmüller and D. Stricker, "Comparison of Kinect V1 and V2 depth images in terms of accuracy and precision," in *Proc. ACCV Workshops*, vol. 10117. Cham, Switzerland: Springer, 2017, pp. 34–45.

[29] Y. Zhang, S. Wang, X. Yang, and Y. Ma, "Robotic vehicle navigation based on image processing using Kinect," in *Proc. Int. Conf. Smart City Syst. Eng. (ICSCSE)*, Hunan, China, Nov. 2016, pp. 18–22, doi: 10.1109/ICSCSE.2016.0015.

[30] A. Basiri, M. A. Oskoei, A. Basiri, and A. M. Shahri, "Improving robot navigation and obstacle avoidance using Kinect 2.0," in *Proc. 5th RSI Int. Conf. Robot. Mechatronics (ICRoM)*, Tehran, Italy, Oct. 2017, pp. 486–489, doi: 10.1109/ICRoM.2017.8466145.

[31] K. Yelamarthi, B. P. DeJong, and K. Laubhan, "A Kinect based vibrotactile feedback system to assist the visually impaired," in *Proc. IEEE 57th Int. Midwest Symp. Circuits Syst. (MWSCAS)*, College Station, TX, USA, Aug. 2014, pp. 635–638, doi: 10.1109/MWSCAS.2014.6908495.

[32] T. Wang, L. Bu, and Z. Huang, "A new method for obstacle detection based on Kinect depth image," in *Proc. Chin. Autom. Congr. (CAC)*, Wuhan, China, May 2015, pp. 537–541, doi: 10.1109/CAC.2015.7382559.

[33] R. Mardiyanto, J. Anggoro, and F. Budiman, "2D map creator for robot navigation by utilizing Kinect and rotary encoder," in *Proc. Int. Seminar on Intell. Technol. Appl. (ISITIA)*, Surabaya, ID, USA, 2015, pp. 81–84, doi: 10.1109/ISITIA.2015.7219957.

[34] R. Lun and W. Zhao, "A survey of applications and human motion recognition with Microsoft Kinect," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 29, no. 5, Aug. 2015, Art. no. 1555008, doi: 10.1142/S0218001415550083.

[35] P. Mittal, R. Singh, and A. Sharma, "Deep learning-based object detection in low-altitude UAV datasets: A survey," *Image Vis. Comput.*, vol. 104, May 2020, Art. no. 104046, doi: 10.1016/j.imavis.2020.104046.

[36] M. S. Alam and J. Oluoch, "A survey of safe landing zone detection techniques for autonomous unmanned aerial vehicles (UAVs)," *Expert Syst. Appl.*, vol. 179, May 2021, Art. no. 115091, doi: 10.1016/j.eswa.2021.115091.

[37] S. B. Kotsiantis, "Feature selection for machine learning classification problems: A recent overview," *Artif. Intell. Rev.*, vol. 42, no. 1, pp. 157–176, 2011, doi: 10.1007/s10462-011-9230-1.

[38] K. M. Veena, S. K. Manjula, and K. B. A. Shenoy, "Performance comparison of machine learning classification algorithms," in *Communications in Computer and Information Science*. Singapore: Springer, 2018, pp. 489–497, doi: 10.1007/978-981-13-1813-9_49.

[39] M. G. Wollsen, J. Hallam, and B. N. Jorgensen, "Novel automatic filter-class feature selection for machine learning regression," in *Advances in Big Data*. Cham, Switzerland: Springer, 2016, pp. 71–80, doi: 10.1007/978-3-319-47898-2_8.

[40] J. Garcia-Gutierrez, A. Troncoso, and J. C. Riquelme, "A comparative study of machine learning regression methods on LiDAR data: A case study," in *Advances in Intelligent Systems and Computing*. Cham, Switzerland: Springer, 2014, pp. 249–258, doi: 10.1007/978-3-319-01854-6_26.

[41] T. Jebara, *Machine Learning*. Boston, MA, USA: Springer, 2004, doi: 10.1007/978-1-4419-9011-2.

[42] T. Karras, S. Laine, and T. Aila, "A style-basedgenerator architecture for generative adversarial networks," in *Proc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2019, pp. 4401–4410.

[43] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, Oct. 2017, pp. 2242–2251, doi: 10.1109/ICCV.2017.244.
[44] M. El-Kaddoury, A. Mahmoudi, and M. M. Himmi, "Deep generative models for image generation: A practical comparison between variational autoencoders and generative adversarial networks," in *Mobile, Secure, Programmable Networking* (Lecture Notes in Computer Science), vol. 11557, E. Renault, S. Boumerdassi, C. Leghris, and S. Bouzefrane, Eds. Cham, Switzerland: Springer, 2019, pp. 1–8, doi: 10.1007/978-3-030-22885-9_1.
[45] O. Press, A. Bar, B. Bogin, J. Berant, and L. Wolf, "Language generation with recurrent generative adversarial networks without pre-training," 2017, *arXiv:1706.01399*.
[46] L. Yu, W. Zhang, J. Wang, and Y. Yu, "SeqGAN: Sequence generative adversarial nets with policy gradient," in *Proc. AAAI*, 2017, pp. 2852–2858.
[47] A. Varde, E. Rundensteiner, G. Javidi, E. Sheybani, and J. Liang, "Learning the relative importance of features in image data," in *Proc. IEEE 23rd Int. Conf. Data Eng. Workshop*, Istanbul, Turkey, Apr. 2007, pp. 237–244, doi: 10.1109/ICDEW.2007.4400998.
[48] G. S. Veena, N. D. Venkata, M. M. Goudar, A. P. Sarashetti, and A. Acharya, "A survey on in-domain and cross-domain image classification using surf features," in *Proc. Int. Conf. Adv. Comput., Commun. Informat. (ICACCI)*, Udupi, India, Sep. 2017, pp. 1797–1802, doi: 10.1109/ICACCI.2017.8126105.
[49] C. Dan Marinescu and M. G. Marinescu, "Classical and quantum information theory," in *Classical and Quantum Information*, C. Dan Marinescu and M. G. Marinescu, Eds. New York, NY, USA: Academic, 2012, pp. 221–344, doi: 10.1016/B978-0-12-383874-2.00003-5.
[50] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. 27th Int. Conf. Neural Inf. Process. Syst.*, vol. 2. Cambridge, MA, USA: MIT Press, 2014, pp. 2672–2680.
[51] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1, San Diego, CA, USA, Oct. 2005, pp. 886–893, doi: 10.1109/CVPR.2005.177.
[52] J. Maldonado-Romo, M. Aldape-Pérez, and A. Rodrïguez-Molina, "Path planning generator with metadata through a domain change by GAN between physical and virtual environments," *Sensors*, vol. 21, p. 7667, Oct. 2021, doi: 10.3390/s21227667.
[53] D. Eigen and R. Fergus, "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 2650–2658, doi: 10.1109/ICCV.2015.304.
[54] K. Karsch, C. Liu, and S. B. Kang, "Depth extrac-tion from video using non-parametric sampling," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2012, pp. 775–788.
[55] L. Ladicky, J. Shi, and M. Pollefeys, "Pulling things out of perspective," in *Proc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2014, pp. 89–96.
[56] M. Liu, M. Salzmann, and X. He, "Discrete-continuous depth estimation from a single image," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 716–723.
[57] 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 *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1119–1127.
[58] P. Wang, X. Shen, Z. Lin, S. Cohen, B. Price, and A. Yuille, "Towards unified depth and semantic prediction from a single image," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 2800–2809.
[59] D. Eigen, C. Puhrsch, and R. Fergus, "Prediction from a single image using a multi-scale deep network," in *Proc. Conf. Neural Inf. Process. Syst. (NIPS)*, 2014, pp. 1–5.
[60] A. Roy and S. Todorovic, "Monocular depth estimation using neural regression forest," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 5506–5514.
[61] A. Valade, P. Acco, P. Grabolosa, and J.-Y. Fourniols, "A study about Kalman filters applied to embedded sensors," *Sensors*, vol. 17, no. 12, p. 2810, 2017, doi: 10.3390/s17122810.

JAVIER MALDONADO-ROMO received the B.S. and M.S. degrees from the Instituto Politécnico Nacional of Mexico, in 2013 and 2017, respectively, and a business administration degree from the Universidad Nacional Autónoma de México. He is currently pursuing the Ph.D. degree in robotics and mechatronics systems with the Instituto Politécnico Nacional de México. His research interests include machine learning and new technologies as sustainable strategies to elaborate novel developments.

MARIO ALDAPE-PÉREZ received the Ph.D. degree in computer science from the National Polytechnic Institute of Mexico, in 2011. He was the Head of the Research and Technology Innovation Department, CIDETEC, from 2013 to 2016. He is currently a Professor of computer architecture with CIDETEC, National Polytechnic Institute of Mexico. His current research interests include associative memories, soft computing, and FPGA implementation of high performance pattern classification algorithms. He is a member of the IEEE Computer Society and the ACM.

ALEJANDRO RODRÍGUEZ-MOLINA (Member, IEEE) received the B.S. degree in computer systems engineering from the Escuela Superior de Cómputo (ESCOM), Instituto Politécnico Nacional (IPN), in 2013, the M.Sc. degree in computer science from the Centro de Investigación y de Estudios Avanzados (CINVESTAV), IPN, in 2015, and the Ph.D. degree in robotics and mechatronic systems engineering from the Centro de Innovación y Desarrollo Tecnológico en Cómputo (CIDETEC), IPN, in 2019. He is currently a full-time Professor with the Research and Postgraduate Division, Instituto Tecnológico de Tlalnepantla (ITTLA), Tecnológico Nacional de México (TecNM). His research interests include design and implementation of bio-inspired meta-heuristics for optimization and their application to engineering problems.