2022 IEEE/SICE International Symposium on System Integration (SII)
January 9-12, 2022. Narvik, Norway

Extending HoloGAN by Embedding Image Content into Latent Vectors for Novel View Synthesis

Jing Wang[1], Lotfi El Hafi[1,*], Akira Taniguchi[1], Yoshinobu Hagiwara[1], and Tadahiro Taniguchi[1]

***Abstract*— This study aims to further develop the task of novel view synthesis by generative adversarial networks (GAN). The goal of novel view synthesis is to, given one or more input images, synthesize images of the same target content but from different viewpoints. Previous research showed that the unsupervised learning model HoloGAN achieved high performance in generating images from different viewpoints. However, HoloGAN is less capable of specifying the target content to generate and is difficult to train due to high data requirements. Therefore, this study proposes two approaches to improve the current limitations of HoloGAN and make it suitable for the task of novel view synthesis. The first approach reuses the encoder network of HoloGAN to get the corresponding latent vectors of the image contents to specify the target content of the generated images. The second approach introduces an autoencoder architecture to HoloGAN so that more viewpoints can be generated correctly. The experiment results indicate that the first approach is efficient in specifying a target content. Meanwhile, the second approach method helps HoloGAN to learn a richer range of viewpoints but is not compatible with the first approach. The combination of these two approaches and their application to service robotics are discussed in conclusion.**

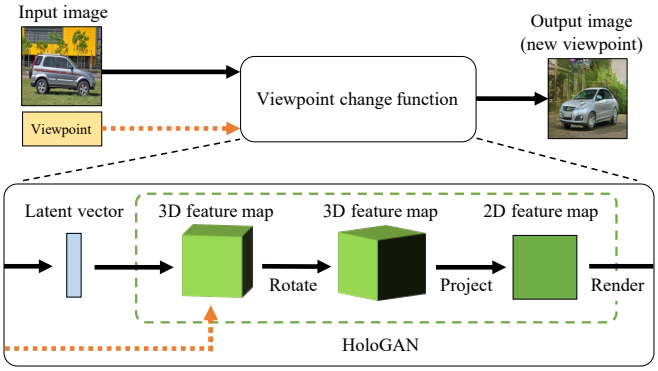


Fig. 1. The purpose of this study is to find a function that takes in an image and a viewpoint as inputs, then generate an image that has the same content but from the input viewpoint. HoloGAN can already generate images from different viewpoints by rotating the learned 3D feature map to a specified angle, but it does not explicitly describe the relationship between the latent vectors and the contents of the generated images. Therefore, we propose to embed input images into latent vectors to specify the content of the generation to achieve novel view synthesis.

I. Introduction

When we, humans, are looking around the world, our retina can be thought of as a 2D screen that receives light and sends 2D images to the brain. However, we can perceive and interact with the 3D world by processing this 2D visual information. For example, one classic experiment [1] demonstrates that we excel at mental rotation, i.e., predicting what a given object would look like after a known 3D rotation is applied. In this study, we attempt to approach the computational equivalent of mental rotation, also called novel view synthesis.

The objective of a novel view synthesis task is to, given one or more images, synthesize images of the same target object or scene but from different arbitrary viewpoints. This technology can benefit many applications, and especially in the field of robotics. For example, it can improve human face recognition [2], help construct 3D models of objects for grasp planning [3], [4], or operate in conjunction with other cameras to see through objects [5].

This study was supported by the Japan Ministry of Education, Culture, Sports, Science and Technology (MEXT) KAKENHI Grant-in-Aid for Scientific Research on Innovative Areas (Area no. 4805, Task no. 16H06569).

[1]Jing Wang, Lotfi El Hafi, Akira Taniguchi, Yoshinobu Hagiwara, and Tadahiro Taniguchi are with Ritsumeikan University; 1-1-1 Noji-Higashi, Kusatsu, Shiga 525-8577, Japan. {wang.jing, lotfi.elhafi, a.taniguchi, yhagiwara, taniguchi}@em.ci.ritsumei.ac.jp

*Corresponding author.

Many methods have been considered for novel view synthesis. They can be roughly divided into two categories: Geometry-based methods [6], [7], [8], [9], [10], and learning-based methods [11], [2], [12].

On the one hand, the geometry-based methods try to first estimate the approximate underlying 3D structure of the target content and then change the viewpoint. While geometry-based methods can synthesize images with high quality, they typically only suit objects with simple shapes or otherwise consume lots of computational resources.

On the other hand, the recent breakthroughs in deep learning allowed the development of learning-based methods to generate new viewpoints. Even though common learning-based methods cannot synthesize images with quality higher than geometry-based methods, they can change viewpoints more broadly even when given only a few visual information as input. In addition, learning-based methods have a higher generalization performance when applying the previously learned knowledge to unseen objects.

One popular learning-based model to generate images with different viewpoints is HoloGAN [13]. It is a generative model which borrows some concepts from the style-based generator of StyleGAN [14]. HoloGAN learns 3D representations directly from 2D natural images in an entirely unsupervised manner and can control the viewpoints of the generated images well. However, HoloGAN is not a sufficient method for novel view synthesis because of the

978-1-6654-4539-9/22/$31.00 ©2022 IEEE

following two issues:

1) HoloGAN can only generate images of random contents by feeding random latent vectors, i.e., it cannot control the target content of the generated images.
2) HoloGAN is sensitive to its hyperparameters and difficult to train, which often results in only learning a partial range of the viewpoints contained in the dataset.

Especially, 1) is crucial, as the goal of novel view synthesis is to change the viewpoint of given images or contents. Meanwhile, the difficulty of training described in 2) also limits the performance and adoption of HoloGAN for novel view synthesis applications. Therefore, this study aims to extend the HoloGAN by addressing these two issues. An overview of this purpose is shown in Fig. 1.

We propose one approach with two different methods to solve 1) and one approach with a single method to solve 2). More precisely, the first and second methods described in the following paragraphs solve 1), while the third method solves 2).

First, we reuse the encoder network of HoloGAN to embed images into latent vectors. Inspired by StyleALAE [15], we assume that the control over input latent vectors determines the contents of the generated images. The encoder network of HoloGAN will embed the target content into a corresponding latent vector. Using the latent vector, it is possible to apply HoloGAN to generate images of the same content from different viewpoints.

Second, to improve the first method, we update the embedded latent vectors to find better representative latent vectors of the target contents. Image2StyleGAN [16] and iGAN [17] suggest that updating the embedded latent vectors to reduce the difference between the target contents in the input images and the generated images leads to better performances. We also adopt this idea.

Third, we investigate the addition of new criteria with the introduction of an auto-encoder architecture for HoloGAN to learn a broader range of viewpoints.

In summary, our contribution to novel view synthesis with this study is an extension of HoloGAN that allows to:

- Specify the target contents of the generated images without sacrificing the quality by introducing control over the latent vectors.
- Generate images of the target content with a broader range of viewpoints by introducing an auto-encoder architecture.

The remainder of this paper is structured as follows. Section II introduces works related to novel view synthesis. Section III describes the methods proposed to extend HoloGAN for novel view synthesis. Section IV details the experiments conducted to evaluate the proposed extensions. Section V discusses the performances and limitations observed in the experimental results. Finally, Section VI concludes this paper with avenues for future works.

II. Related Works

The approaches to achieve novel view synthesis can be broadly divided into two categories: geometry-based and learning-based.

On the one hand, the geometry-based methods try to first estimate the approximate underlying 3D structure of a target object and then change the viewpoint. Early methods developed at the end of the 20th include view morphing [6] and image reprojection [7]. Since then, newer geometry-based methods have reached high performance. However, they either assume that the target object only has simple shapes, such as cylinders and spheres [8], or that the target object has to be observed from different viewpoints to collect large amounts of visual 3D information beforehand [9], [10]. In summary, while geometry-based methods can generally synthesize high-quality images, they typically only suit objects with simple shapes or otherwise consume lots of computational resources.

On the other hand, the recent breakthroughs in deep learning allowed the development of learning-based methods to generate new viewpoints by, for example, predicting the appearance flow [11], generating images from pixels [2], or reconstructing meshes of objects [12]. Although most learning-based methods cannot synthesize high-quality images compared to geometry-based methods, they have other advantages. Indeed, learning-based methods typically learn functions to predict different views of the target content in the input image by processing pre-existing data. They can change viewpoint broadly even when only given a few visual information, while geometry-based methods need visual information or point cloud from various viewpoints for every synthesis to address potential occlusions. However, although learning-based methods are promising, they still have drawbacks. In particular, even if learning-based methods do not need much data when synthesizing, they usually require large amounts of pre-existing data with either precise labels or synthetic 3D models which are both time-consuming and labor-intensive.

A. HoloGAN

With similar considerations regarding the data issue, Nguyen-Phuoc et al. proposed HoloGAN [13], a generative model that can achieve state-of-the-art performance in generating images of various viewpoints by imitating rendering techniques of computer graphics with a neural network that can be trained in an entirely unsupervised manner. HoloGAN first generates a 3D representation of a target content and then projects it to a 2D representation in order to generate, or "render", an image. The rotation applied to the 3D representation directly leads to the change of viewpoints. As a result, HoloGAN can explicitly control the viewpoints of the generated images. Considering the lesser requirements on the dataset and the explicit control on viewpoints, we assume that HoloGAN can be extended for novel view synthesis.

B. Latent Space Embedding

In general, there are two approaches to embed contents from the image space to the latent space: One is training an encoder that maps a given image to the latent space, e.g., Variational Auto-Encoder [18], while the other is selecting

a random initial latent code and optimizing it using gradient descent [17], [16].

In the first approach, the encoder neural network should be designed correspondingly to the generator, as the encoder can be thought of as an inverse function of the generator. Therefore, style-based generators that have a different architecture than traditional generators, like the one of StyleGAN [14], may need an encoder built with a corresponding architecture. Under this concept, Pidhorskyi et al. proposed StyleALAE [15] which further developed StyleGAN with the introduction of an auto-encoder while keeping almost the same quality of generated images.

On the other hand, the second approach assumes a pre-trained generator, so that training an encoder and a generator together from scratch is not required. Hence, designing an encoder for the second approach is easier than for the first approach. Abdal et al. proposed Image2StyleGAN [16] which uses the second approach for latent space embedding with style-based generators.

Because one of the aims of this study is to control the synthesized contents of HoloGAN, and because HoloGAN has a style-based generator, both StyleALAE, i.e., the first approach, and Image2StyleGAN, i.e., the second approach, are worth considering. Fortunately, HoloGAN already has a generator-encoder architecture which means that the first approach can be implemented easily. In addition, this study also investigates the second approach with iGAN [17] that uses the output of an encoder network as the initial latent vector.

III. Proposed Methods

This section formally describes the main objective of generating images with specific content from different viewpoints and the proposed methods to approach it by extending HoloGAN. The symbol notations are summarized in Table I.

TABLE I

SYMBOLS DESCRIBING THE PROPOSED METHODS.

Symbols	Meanings
x	Image.
x^R	Real image.
x^*	Generated image.
x^*_θ	Generated image from θ.
x^*_{rec}	Reconstructed image.
θ	Viewpoint.
θ^*	Output viewpoint.
z	Latent vector.
z^*	Output latent vector.
w^+	Transformed latent vector.
w^{+*}	Transformed latent vector used in our methods.
$f(\cdot)$	Function for novel view synthesis.
$G(\cdot)$	Generator.
$D(\cdot)$	Discriminator.
$D_i(\cdot)$	Neural network of D until the last but i layer.
$F(\cdot)$	Encoder (identity regularizer of HoloGAN).
$A(\cdot)$	Learned affine transformation.
$M_1(\cdot)$	Function of Method 1.
$M_2(\cdot)$	Function of Method 2.
$L(\cdot)$	Distance metric for the difference between images.
$L_f(\cdot)$	Feature-level loss function.
$L_{view}(\cdot)$	Reconstruction loss.

A. Main Objective

Assuming that we have an image x, the goal is to find a function f to generate images of the same content than x but from different viewpoints, as summarized by Equation (1). Here, x_θ denotes a generated image that has the same content as x but seen from the target viewpoint θ.

$$x_\theta = f(x, \theta) \tag{1}$$

HoloGAN provides a suitable model for this purpose. Equation (2) shows the generation process of HoloGAN where x^*_θ is the generated image from the target viewpoint θ, z is the latent vector, w^+ is the transformed latent vector, G is the generation function learned by HoloGAN, and A is a learned affine transformation. It indicates that HoloGAN can generate images of the same content but from different viewpoints by changing θ while keeping z. For example, $x^*_{\theta_1} = G(A(z), \theta_1)$ and $x^*_{\theta_2} = G(A(z), \theta_2)$ should have the same content but observed from the two different viewpoints θ_1 and θ_2, respectively.

$$x^*_\theta = G(A(z), \theta) \tag{2}$$

$$= G(w^+, \theta) \tag{3}$$

However, HoloGAN does not explicitly describe the relationship between the latent vector z and the content of the generated image x^*_θ. Due to this, it is difficult to feed an image as the target content during generation, thus preventing novel view synthesis. Therefore, we need to find a function M which embeds the image x to a suitable latent vector w^{+*}, as expressed by Equation (4). In particular, we approximate the embedding function M is by M_1 and M_2 that correspond respectively to the Method 1 and Method 2 proposed in this study.

$$w^{+*} = M(x) = M_2(A(M_1(x))) \tag{4}$$

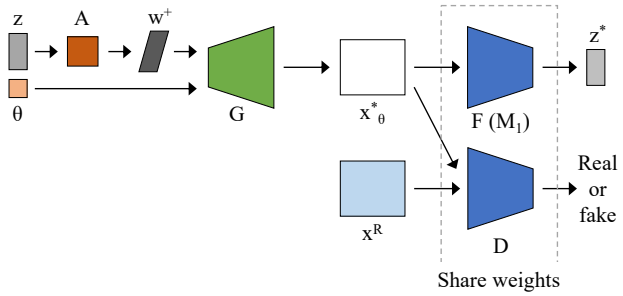
B. Method 1: Reusing the Encoder of HoloGAN

Assuming that HoloGAN provides a capable pre-trained generator G, the objective of this method is to find an embedding function M_1 that makes use of an encoder network to map the target content from the image space to the latent space. Fig. 2a gives an overview of the HoloGAN architecture.

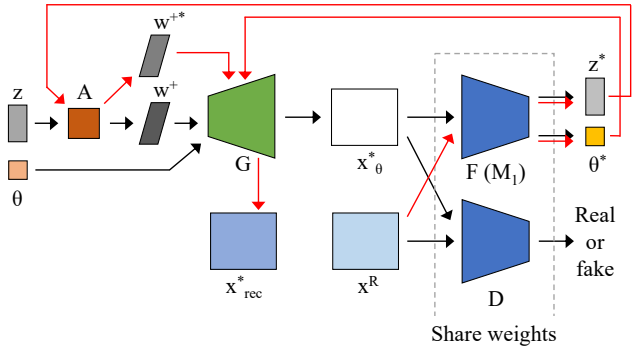
We propose to reuse the identity regularizer network of HoloGAN, denoted F, as the encoder for this purpose. F takes the generated image x^*_θ and outputs the corresponding latent vector z^*. This process is similar to an encoder network, so we refer to F as an encoder. Based on this observation, we can first get an embedded latent vector z^* by feeding a target image x to the encoder network F, as shown in Equation (5). Here, the encoder network F is corresponding to function M_1 in Equation (4).

$$z^* = F(x) \tag{5}$$

Next, we can feed the embedded latent vector z^* with a viewpoint parameter θ to the generator G to output images of the same content as the target image x but from the viewpoint



(a) Overview of the architecture of HoloGAN. The inputs of the generator G are the viewpoint θ and latent vector z. z is fed to an affine transformation unit A to get the transformed latent vector w^+. The generated image x^*_θ and real image x^R are classified by the discriminator D. Note that the special discriminator D_L of HoloGAN is omitted here for better visualization. The identity regularizer F, referred to as an encoder in this study, shares weights with D. F maps x^*_θ to a latent vector z^* as close as possible to z. F is also referred to as M_1 in this study.



(b) Overview of the architecture of HoloGAN+AE. The differences with the original architecture of HoloGAN are highlighted in red. In HoloGAN+AE, F not only learns z^* but also learns the viewpoint θ^* and makes it as close as possible to θ. The flow of the additional auto-encoder architecture is $x^R \to F \to (z^*, \theta^*) \to G \to x^*_{rec}$.

Fig. 2. Overview of the architectures of HoloGAN and HoloGAN+AE.

θ, as described by Equation (6).

$$x^*_\theta = G(A(z^*), \theta) \tag{6}$$

C. Method 2: Updating the Embedded Latent Vector

Limited by the capacity of the encoder network F, sometimes the embedded latent vectors cannot sufficiently represent the contents of the target images. For example, the color of the target object in the input image is purple, but the generated result is red. Therefore, we propose to find more representative latent vectors by updating the embedded latent vectors. Similarly to Method 1, this second method also assumes that HoloGAN provides a capable pre-trained generator G.

As for the choice of the latent space, Image2StyleGAN suggests that the most efficient way is to update latent vectors in the latent space $\mathbb{W}^+$ of w^+, thus we also choose to focus on the latent space $\mathbb{W}^+$, as shown in Equation (7) where Method 2 is denoted by the function M_2.

$$w^{+*} = M_2(A(z^*)) \tag{7}$$

The update aims to reduce the difference, or loss, between the input image and the reconstructed image. The reconstructed image here means image generated by feeding the embedded latent vector. In an ideal case, we can assume that a real image x^R lies on a manifold $\mathbb{M} \subseteq \mathbb{X}$, where $\mathbb{X}$ denotes the image space. The generator of HoloGAN can approximate the manifold $\mathbb{M}$ by training, but the approximated manifold is not equal to $\mathbb{M}$, so we denote the approximated manifold as $\tilde{\mathbb{M}}$. Here, the goal of finding a more representative latent vector is similar to finding an image $x^* \in \mathbb{X}$ which is close to the real x^R one, as expressed by Equation (8). $L(x_1, x_2)$ is a distance metric that can be used to measure the distance between the two images x_1 and x_2. We use Equation (9) as $L(x_1, x_2)$ in this study, where N_p is the number of scalars of the input image, i.e., $N_P = n \times n \times 3$. In the loss function $L_f(x_1, x_2)$, D_3 is a network that reuses the convolutional layers of D until the last but two layers, while D_4 reuses layers until the last but one layer. N_i is the number of scalars in the output of D_i.

$$x^* = \arg\min_{x \in \tilde{\mathbb{M}}} L(x, x^R) \tag{8}$$

$$L(x_1, x_2) = L_f(x_1, x_2) + \frac{1}{N_p}\|x_1 - x_2\|_2^2 \tag{9}$$

$$L_f(x_1, x_2) = \sum_{i=3}^{4} \frac{1}{N_i}\|D_i(x_1) - D_i(x_2)\|_2^2 \tag{10}$$

As a result, we propose to update w^{+*} and θ^* to find more representative latent vectors, as described in Equations (11) and (12).

$$w^{+*} \leftarrow w^{+*} - \frac{\partial L(G(w^{+*}, \theta^*), x^R)}{\partial w^{+*}} \tag{11}$$

$$\theta^* \leftarrow \theta^* - \frac{\partial L(G(w^{+*}, \theta^*), x^R)}{\partial \theta^*} \tag{12}$$

D. Method 3: Introducing an Auto-Encoder

Both Method 1 and Method 2 were built on the assumption that HoloGAN provides a capable generator G. However, HoloGAN is sensitive to its hyperparameters and difficult to train, which often results in only learning a partial range of the viewpoints contained in the dataset. Therefore, we propose to introduce an auto-encoder architecture to HoloGAN so that it can be trained with additional criteria for improved stability and broader viewpoints. An overview of this method is shown in Fig. 2b.

First, we make the encoder network F learn embedded latent vectors and viewpoints simultaneously, as expressed by Equation (13). In addition, we consider the loss function described in Equation (14), where N is the number of data, to minimize the difference between the learned viewpoint θ^* and the input viewpoint θ.

$$(z^*, \theta^*) = F(x) \tag{13}$$

$$L_{view} = \frac{1}{N}\sum_{i=1}^{N}\|\theta^*_i - \theta_i\|_2^2 \tag{14}$$

Finally, we make an auto-encoder by using the encoder network F as the encoder and the generator network G as the

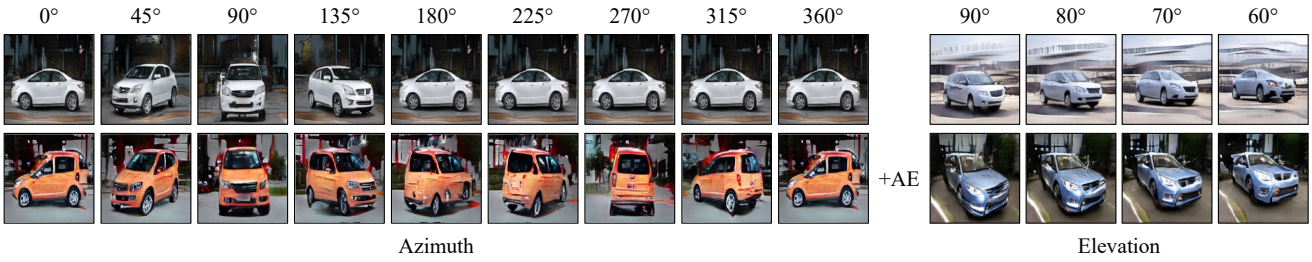


Fig. 3. Samples of images generated with HoloGAN (top) and HoloGAN+AE (bottom) when fed with the random latent vector z. The azimuth viewpoints range from 0° to 360°. Note that the car should face left at azimuth 0° and front at azimuth 90°. The elevation viewpoints range from 60° to 90°.

Fig. 4. The 5 input images used in Experiment 2.

decoder. The real image x^R is fed to the encoder F to get the embedded latent vector z^* and the corresponding viewpoint θ^*. z^* and θ^* are then passed to the generator network G to reconstruct the real image, as described by Equation (15). The reconstruction loss is defined by Equation (16), where N is the number of data.

$$x^*_{rec} = G(F(x^R)) \tag{15}$$

$$L_{rec} = \frac{1}{N}\sum_{i=1}^{N} \|x^*_{rec,i} - x^R_i\|_2^2 \tag{16}$$

We further refer to this modified HoloGAN architecture as HoloGAN+AE.

IV. Experiments

This section details the experiments conducted to evaluate the proposed methods. We made the code that implements these experiments available online[1] to reproduce our results with ease.

A. Dataset and Hyperparameters

We use the Compcar [19] dataset in our experiments. Compcar includes $136,726$ natural images of cars taken from different viewpoints. We used the bounding boxes provided in the dataset to dispose of the non-centered images. As a result, $130,084$ images in total were retained for the experiment. The images were further divided into $120,084$ training data and $10,000$ testing data.

Regarding the training HoloGAN, we set the hyperparameters as follows. The batch size was 32. The dimension of the input latent vector z was 200. The resolutions of the input and output images were both 128×128 pixels. The total number of training epochs was 25 with the learning rate η linearly decreasing from 0.00005 until epoch 12 to 0 after epoch 25. The optimizers used for all variables was Adam with $\beta_1 = 0.5$ and $\beta_2 = 0.999$. The azimuth angle ranged from 0° to 359°. The elevation angle ranged from 60° to 90°. The scaling ranged from 0.8 to 1.3. Note that we used the same set of hyperparameters to train HoloGAN+AE.

Regarding the optimization of the embedded latent vectors, we used gradient descent. The learning rate was 2.0, and the number of iterations was 3.

[1] https://github.com/xxxiaojing/extending-hologan-for-novel-view-synthesis

B. Experiment 1: Generate Images by Feeding Random Latent Vectors

This experiment is designed to compare the ability of HoloGAN and HoloGAN+AE to generate images from different viewpoints by feeding random latent vectors. We generated images ranging from an azimuth of 0° to 360° and an elevation from 60° to 90° by the feeding random latent vector z to both HoloGAN and HoloGAN+AE.

Fig. 3 shows the results with samples of the same latent vectors but from different azimuth (left side) and elevation (right side) viewpoints. It appears that HoloGAN could only generate viewpoints with an azimuth range from 0° to 180°. Over 180° of azimuth, the generated images were identical to the images of 0° of azimuth. Whereas HoloGAN+AE could generate viewpoints with an azimuth range from 0° to 360°, the viewpoints around 180° of azimuth were not learned well. On the other hand, both HoloGAN and HoloGAN+AE could generate elevation viewpoints ranging from 60° to 90°.

C. Experiment 2: Evaluate the Mean Opinion Score of the Generated Images

The purpose of this experiment is to compare the ability of HoloGAN and HoloGAN+AE to generate target content at specific viewpoints when given an input image, i.e., achieving novel view synthesis. 4 combinations of the proposed methods were considered:

1) *HoloGAN embed* (baseline): Feed the embedded latent vectors z^* of the target contents to HoloGAN (Method 1).
2) *HoloGAN embed+update*: Feed the updated embedded latent vectors w^{+*} of the target contents to HoloGAN (Methods 1+2).
3) *HoloGAN+AE embed*: Feed the embedded latent vectors z^* of the target contents to HoloGAN+AE (Methods 1+3).
4) *HoloGAN+AE embed+update*: Feed the updated embedded latent vectors w^{+*} of the target contents to HoloGAN+AE (Methods 1+2+3).

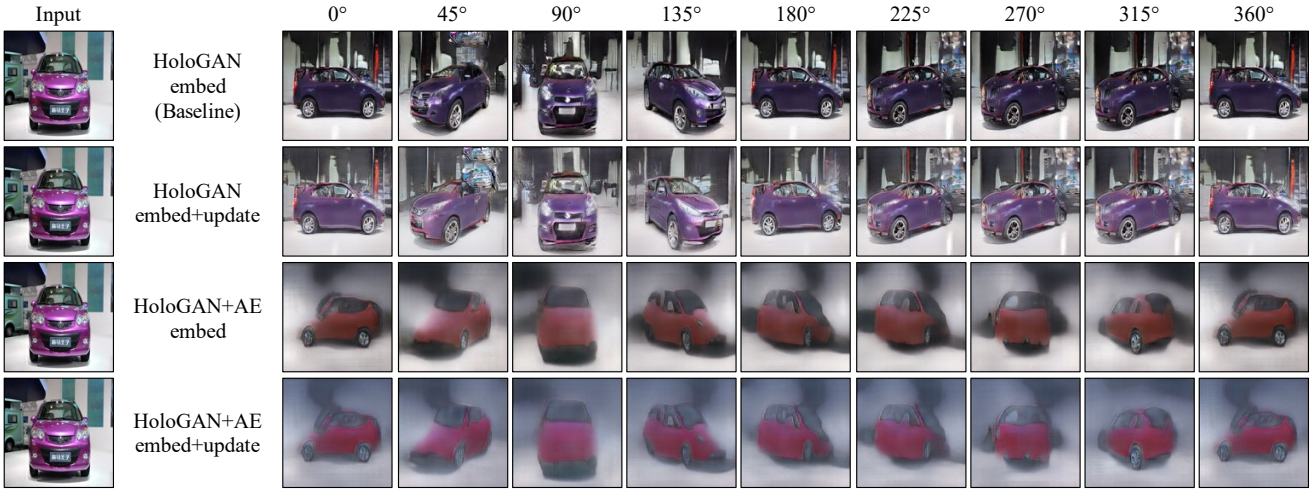


Fig. 5. Examples of images generated in Experiment 2. From a real input image (left), 8 viewpoints (right) ranging from azimuth 0° to 360° by steps of 45° were generated using each of the 4 combinations of proposed methods. Each line constitutes an experimental set to be subsequently assessed by MOS.

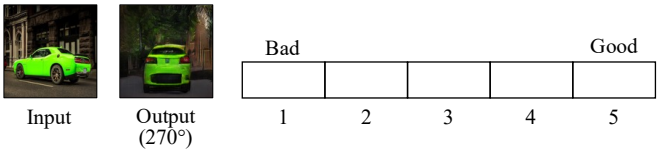


Fig. 6. Examples of samples used for the MOS assessment. The subjects were asked to score the quality of the generated images on a scale from 1 to 5, considering the similarity of the target content and accuracy of the viewpoint angle. 1 indicates bad quality, while 5 indicates good quality.

5 input images were randomly sampled from the testing data for this experiment. They are shown in Fig. 4. For each of these input images, 8 viewpoints ranging from azimuth 0° to 360° by steps of 45° were generated using each of the 4 combinations of proposed methods. The results are shown in Fig. 5.

We use the mean opinion score (MOS) to assess the quality of the generated images. MOS is a numerical measure of the human-judged overall quality of a stimulus or system. Specifically, given the input image and corresponding generated images from new viewpoints, the subjects were asked to score the quality of the generated images on a scale from 1 to 5 while considering the similarity of the target content and accuracy of the viewpoint angle, as shown in Fig. 6. 8 subjects participated in the experiments. Each of them assessed the quality of 20 sets of 8 generated viewpoints: 5 input images times 4 combinations of proposed methods. To reduce bias, the 20 sets were arranged in random order.

The results of the MOS assessment are shown in Fig. 7. *HoloGAN embed+update* performed better than *HoloGAN embed*, while *HoloGAN+AE embed+update* performed better than *HoloGAN+AE embed*. This indicates that the updating process of the embedded latent vectors is effective. However, the results did not show significant differences in the Welch's t-test. As for *HoloGAN+AE embed* and *HoloGAN+AE embed+update*, which adopt the auto-encoder, both of them scored less than the architectures without an auto-encoder. The significant blurriness of the generated images may be to blame.

V. Discussion

When trained on the Compcar dataset, the original HoloGAN paper [13] presented results of a full range of generated azimuth viewpoints, whereas we found in our experiment that it was not stable for novel view synthesis and could only generate a partial range of azimuth viewpoints, as previously seen in Fig. 3. Compared to HoloGAN, HoloGAN+AE could generate a broader range of azimuth viewpoints. Thus HoloGAN+AE appears more efficient to learn viewpoints.

However, in Experiment 2, which assessed the ability to synthesize novel views with the 4 combinations of our proposed methods, the modified HoloGAN architecture outperformed HoloGAN+AE on the MOS assessment. Indeed, when given input images, HoloGAN+AE always generated blurry images, as previously seen in Fig. 5. As clear shapes and outlines are important quality factors, the experiment subjects could hardly recognize a car or the viewpoint in the blurry images. Although HoloGAN+AE can generate images with a richer range of viewpoints when be fed with random latent vectors, it cannot generate the target contents from the specified viewpoints as accurately as HoloGAN. Regarding the reason for the blurriness, the encoder network F of HoloGAN+AE tends to focus on minimizing the pixel-level loss of Equation (16) at the expense the fidelity. In addition, if F does not learn a viewpoint correctly, the reconstruction of the real image is inaccurate, so an experiment that evaluates the learned viewpoint should be further considered.

On the other hand, the original HoloGAN cannot specify the contents of the generation. Therefore, Method 1, which reuses the encoder network of HoloGAN to embed images, has proven to be efficient and necessary for novel view synthesis. In addition, Method 2, which updates the embedded latent vectors, led to better results in Experiment 2. For

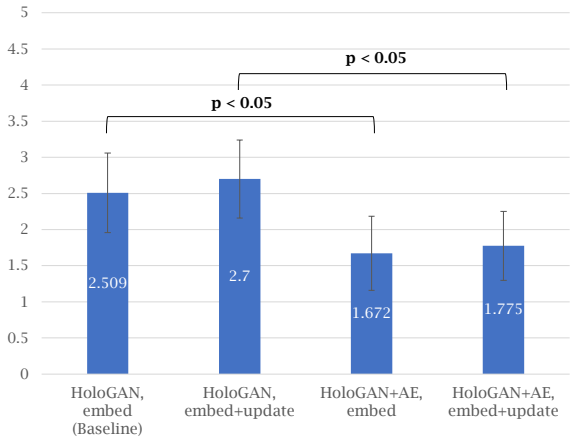


Fig. 7. Results of the MOS assessment of the quality of the generated images by the 4 combinations of proposed methods.

example, we can observe that the generated colors were more similar to the colors of the input images, as demonstrated in Fig. 5. This may be the reason why these generated images scored higher on MOS. However, the Welch's t-test did not show a significant difference, so further experiments are required to evaluate the effectiveness of Method 2.

VI. Conclusion

This study explored the extension of HoloGAN for the task of novel view synthesis. We proposed and evaluated two approaches: 1) controlling the content of the generated images by using an encoder network to embed the target contents into latent vectors and by updating the embedded latent vectors to find more representative ones, and 2) adding new criteria with the introduction of an auto-encoder to make HoloGAN learn a broader range of viewpoints.

The experiments showed that feeding the embedded latent vectors is efficient in specifying the content of the generated images. Moreover, the introduction of the auto-encoder architecture and additional criteria helped HoloGAN generate a richer range of viewpoints. However, the two proposed approaches were not compatible: The images generated from the embedded latent vectors were significantly blurrier when HoloGAN was trained with additional criteria.

Finding a solution to combine these two approaches could be an interesting avenue for future work. In addition, recent works on BlockGAN [20] suggest that GAN-based multiple-object view manipulations could be applied to further extend this study. Finally, improving the resolution of the generated images borrowing inspiration from the StyleGAN [14] architecture is also a direction worth considering.

Finally, we want to apply our proposed methods in service robotics for improving robots' 3D space awareness and object representation to solve the practical issue of occlusion during the planning and grasping in cluttered spaces.

References

[1] R. N. Shepard and J. Metzler, "Mental Rotation of Three-Dimensional Objects," *Science*, vol. 171, no. 3972, pp. 701–703, Feb. 1971.

[2] R. Huang, S. Zhang, T. Li, and R. He, "Beyond Face Rotation: Global and Local Perception GAN for Photorealistic and Identity Preserving Frontal View Synthesis," in *2017 IEEE International Conference on Computer Vision*, Venice, Italy, Oct. 2017, pp. 2458–2467.

[3] J. Varley, C. DeChant, A. Richardson, J. Ruales, and P. Allen, "Shape Completion Enabled Robotic Grasping," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vancouver, Canada, Sept. 2017, pp. 2442–2447.

[4] C. B. Choy, D. Xu, J. Gwak, K. Chen, and S. Savarese, "3D-R2N2: A Unified Approach for Single and Multi-View 3D Object Reconstruction," in *14th European Conference on Computer Vision*, vol. 9912, Amsterdam, Netherlands, Oct. 2016, pp. 628–644.

[5] F. Rameau, H. Ha, K. Joo, J. Choi, K. Park, and I. S. Kweon, "A Real-Time Augmented Reality System to See-Through Cars," *IEEE Transactions on Visualization and Computer Graphics*, vol. 22, no. 11, pp. 2395–2404, Nov. 2016.

[6] S. M. Seitz and C. R. Dyer, "View Morphing," in *23rd ACM Special Interest Group on Computer Graphics and Interactive Techniques*, New Orleans, United States, Aug. 1996, pp. 21–30.

[7] L. McMillan and S. Gortler, "Image-based Rendering: A New Interface between Computer Vision and Computer Graphics," *ACM SIGGRAPH Computer Graphics*, vol. 33, no. 4, pp. 61–64, Nov. 1999.

[8] T. Chen, Z. Zhu, A. Shamir, S.-M. Hu, and D. Cohen-Or, "3-Sweep: Extracting Editable Objects from a Single Photo," *ACM Transactions on Graphics*, vol. 32, no. 6, pp. 1–10, Nov. 2013.

[9] M. Levoy, J. Ginsberg, J. Shade, D. Fulk, K. Pulli, B. Curless, S. Rusinkiewicz, D. Koller, L. Pereira, M. Ginzton, S. Anderson, and J. Davis, "The Digital Michelangelo Project: 3D Scanning of Large Statues," in *27th ACM Special Interest Group on Computer Graphics and Interactive Techniques*, New Orleans, United States, July 2000, pp. 131–144.

[10] A. Haleem, P. Gupta, S. Bahl, M. Javaid, and L. Kumar, "3D Scanning of a Carburetor Body using COMET 3D Scanner supported by COLIN 3D Software: Issues and Solutions," *Materials Today: Proceedings*, vol. 39, no. 1, pp. 331–337, Aug. 2021.

[11] T. Zhou, S. Tulsiani, W. Sun, J. Malik, and A. A. Efros, "View Synthesis by Appearance Flow," in *14th European Conference on Computer Vision*, vol. 9908, Amsterdam, Netherlands, Oct. 2016, pp. 286–301.

[12] C. Wen, Y. Zhang, Z. Li, and Y. Fu, "Pixel2Mesh++: Multi-View 3D Mesh Generation via Deformation," in *2019 IEEE/CVF International Conference on Computer Vision*, Seoul, South Korea, Oct. 2019, pp. 1042–1051.

[13] T. Nguyen-Phuoc, C. Li, L. Theis, C. Richardt, and Y.-L. Yang, "HoloGAN: Unsupervised Learning of 3D Representations from Natural Images," in *2019 IEEE/CVF International Conference on Computer Vision Workshop*, Seoul, South Korea, Oct. 2019, pp. 2037–2040.

[14] T. Karras, S. Laine, and T. Aila, "A Style-based Generator Architecture for Generative Adversarial Networks," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Long Beach, United States, June 2019, pp. 4396–4405.

[15] S. Pidhorskyi, D. A. Adjeroh, and G. Doretto, "Adversarial Latent Autoencoders," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Seattle, United States, June 2020, pp. 14 092–14 101.

[16] R. Abdal, Y. Qin, and P. Wonka, "Image2StyleGAN: How to Embed Images into the StyleGAN Latent Space?" in *2019 IEEE/CVF International Conference on Computer Vision*, Seoul, South Korea, Oct. 2019, pp. 4431–4440.

[17] J.-Y. Zhu, P. Krahenbuhl, E. Shechtman, and A. A. Efros, "Generative Visual Manipulation on the Natural Image Manifold," in *14th European Conference on Computer Vision*, vol. 9909, Amsterdam, Netherlands, Oct. 2016, pp. 597–613.

[18] D. P. Kingma and M. Welling, "Auto-Encoding Variational Bayes," in *2nd International Conference on Learning Representations*, Banff, Canada, Apr. 2014, pp. 1–14.

[19] L. Yang, P. Luo, C. C. Loy, and X. Tang, "A Large-Scale Car Dataset for Fine-Grained Categorization and Verification," in *2015 IEEE Conference on Computer Vision and Pattern Recognition*, Boston, United States, June 2015, pp. 3973–3981.

[20] T. Nguyen-Phuoc, C. Richardt, L. Mai, Y.-L. Yang, and N. Mitra, "BlockGAN: Learning 3D Object-Aware Scene Representations from Unlabelled Images," in *34th Conference on Neural Information Processing Systems*, vol. 33, (Virtual), Dec. 2020, pp. 6767–6778.