Predicting Novel Views Using Generative Adversarial Query Network

Phong Nguyen-Ha[1], Lam Huynh[1], Esa Rahtu[2], and Janne Heikkila[1]

[1] Center for Machine Vision and Signal Analysis, University of Oulu, Finland
[2] Tampere University, Finland
phong.nguyen@oulu.fi

Abstract. The problem of predicting a novel view of the scene using an arbitrary number of observations is a challenging problem for computers as well as for humans. This paper introduces the Generative Adversarial Query Network (GAQN), a general learning framework for novel view synthesis that combines Generative Query Network (GQN) and Generative Adversarial Networks (GANs). The conventional GQN encodes input views into a latent representation that is used to generate a new view through a recurrent variational decoder. The proposed GAQN builds on this work by adding two novel aspects: First, we extend the current GQN architecture with an adversarial loss function for improving the visual quality and convergence speed. Second, we introduce a feature-matching loss function for stabilizing the training procedure. The experiments demonstrate that GAQN is able to produce high-quality results and faster convergence compared to the conventional approach.

Keywords: novel view synthesis · generative adversarial query network · mean feature matching loss

1 Introduction

Humans are easily able to build a mental understanding of the 3D scene based on just 2D images. With such ability, we can effortlessly imagine unseen views of 3D scenes and objects which is currently extremely challenging for computer based systems. Instead of reconstructing the scene as an explicit 3D model, humans can approximate new views by combining images obtained from nearby poses. Such task of predicting an image from a novel view point, given a limited set of other images from the same scene, is referred as a novel view synthesis in computer vision literature.

Novel view synthesis can be considered as a fundamental problem in computer vision and it is still being studied actively by the community. Despite of the tremendous progress obtained during the years, the problem is still far from being solved. There are several reasons making novel view synthesis extremely challenging. First, a perfect solution would require knowledge of the full light field of all visible objects which is usually not possible to obtain due to occlusions and

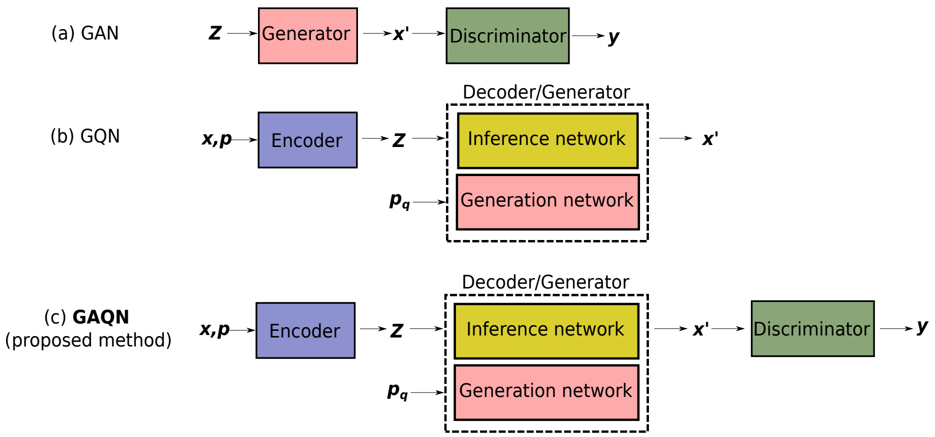

Fig. 1. Illustration of the structure of GAN [9], GQN [7] and our proposed **GAQN** where x, p and p_q are input images, camera poses and queried poses, respectively. z is the latent representation, y is labelled output as real/fake and finally, x' is the generated image.

limited number of samples. Second, due to short distance to the scene geometry, even small changes in the viewpoint may lead to substantial changes in the image appearance. Finally, large parts of the scenes are texture-less and textured areas are typically small. Accurate rendering of novel views can be a useful component for many machine vision applications such as robotics and augmented reality.

The outline of this paper is as follows. In Section 2, we present works related to novel view synthesis methods. A brief background of two previous works that inspired our method is presented in Section 3. In Section 4.1 and Section 4.2, we introduce our Generative Adversarial Query Network with feature matching loss. In Section 5, we show our experimental results and discuss the effectiveness of adversarial training and feature matching loss to the previously proposed methods. Finally, we conclude our research in Section 6.

2 Related work

In this paper, we propose a solution to the task of novel view synthesis when multiple source images are given. Early works on this domain manage to cope with small viewing angle changes by interpolation [5], warping [27] or stereo reconstruction [26]. When the input cameras are further separated from each others, these methods do not perform well due to the sparsity of sampled plenoptic function. Traditional structure-from-motion, structure-from-depth and multi-view geometry methods [4,16,31,34] predict novel views through the estimated 3D structure (point clouds, mesh clouds or a collection of predefined primitives) of the environment. Although, these methods show good results on abundant source data, they are unable to recover the desired target views with a limited number of input images due to the ambiguity of 3D environments. Moreover, estimating the full 3D scene structure may be more challenging problem than that of synthesizing novel images from new viewpoints.

Recent works on deep generative model such as Variational Auto Encoder (VAE) [19], Generative Adversarial Network (GAN) [9] and their variants [1,17,20,33] have demonstrated remarkable results in generating highly photo-realistic novel images. Based on these results, one could expect that similar architectures would be applicable to predict 3D structure of the environment. However, the results demonstrated so far, are far from the desired quality. For instance, the viewpoint transformation networks explicitly learn the relationship between input views of the same 3D scene, but the result is limited in scale such as predicting novel views of an individual rotated objects [6,12,28,29,32,36] or predicting a small displacement between stereo camera views [8,35].

A recent generative model that has shown promises in learning representation for 3D scenes structure is Generative Query Network (GQN) [7]. GQN makes use of an iterative latent variable density model [10] to generate images of the 3D scene. Using multiple source images as input, GQN presents an end-to-end learning framework that generates the novel view from the queried pose by leveraging learned knowledge of the 3D scene representation. However, GQN is known for large memory consumption and the predicted novel views are sometimes blurry.

3 Background

In this section, we provide the reader with a brief background of Generative Adversarial Networks (GANs) [9] and Generative Query Network (GQN) [7]. Figure 1 shows the overall structure of our method and compares it to the previously proposed GANs and GQN.

3.1 Generative Adversarial Networks (GANs)

Generative Adversarial Networks (GANs) [9] consist of two competing architectures referred as a generator (G) and a discriminator (D) (see Figure 1(a)). The generator G maps a given latent representation z (possibly a vector with random values) into a novel image $x' = G(z)$ that is then passed to the discriminator network D. The discriminator aims to determine if the given sample is produced by the generator, or if it is a real image taken from the training set. Denoting the real training samples as x, the conventional generator loss $\mathcal{L}_G^{GAN}$ and discriminator loss $\mathcal{L}_D^{GAN}$ are defined as:

$$\mathcal{L}_G^{GAN} = -\mathbb{E}_{\boldsymbol{z}\sim P_{\boldsymbol{z}}}[\log D(G(\boldsymbol{z}))] \tag{1}$$

$$\mathcal{L}_D^{GAN} = -\mathbb{E}_{\boldsymbol{x}\sim P_x}[\log D(\boldsymbol{x})] - \mathbb{E}_{\boldsymbol{z}\sim P_{\boldsymbol{z}}}[\log(1 - D(G(\boldsymbol{z}))] \tag{2}$$

Both networks are trained simultaneously in an alternating fashion. In the ideal case, the procedure guides the generator to produce images that are indistinguishable from the training image distribution. However, in practice the training procedure is challenging due to various problems such as mode collapses [21].

3.2 Generative Query Network (GQN)

Generative Query Network (GQN) [7] is a deep generative model that learns the scene representation to perform novel view synthesis. Using an arbitrary number of observations, GQN can be trained to generate new views from the same environment. A GQN network includes a scene encoder(*Enc*) and a decoder(*Dec*) as can be seen in Figure 1(b).

First, the *Enc* network tries to compress all of the 3D scene information into a latent representation z from multiple pairs of input views x and their camera poses p. The *Enc* network processes each pair of input views and camera poses by using a feed-forward deep convolutional neural network. Each camera pose p is represented by a 7 dimensional vector of x_t, y_t, z_t, $sin(yaw)$, $cos(yaw)$, $sin(pitch)$ and $cos(pitch)$ where x_t, y_t, z_t are 3D translation, and yaw and $pitch$ are parameters of the 3D rotation matrix. The latent scene information z is the summation of all output pairs from the *Enc* network.

The goal of the GQN is to predict novel views by using the queried pose p_q. Therefore, p_q and z are input to the *Dec* network to generate the new view x'. The *Dec* network is a conditional latent variable model DRAW [11] which includes M pairs of Inference and Generation recurrent sub network [10]. In the language of GANs, the term of decoder *Dec* and generator G have many similarities since they both synthesize data from the latent vector z (could be also random vector). Both *Enc* and *Dec* are trained jointly to minimize the evidence lower bound (ELBO) loss $\mathcal{L}_{GQN}$:

$$\mathcal{L}_{GQN} = -\mathbb{E}_{(x,p),z}\Bigg[- \ln \mathcal{N}(x_{gt}|x') + \sum_{n=1}^{M} \Big[\mathcal{N}(q_m, \pi_m)\Big]\Bigg] \tag{3}$$

The GQN training loss $\mathcal{L}_{GQN}$ is the expected value over the negative log-likelihood of the target image x_{gt} given the target distribution regularized by the cumulative Kullback Leibler divergence between obtained posterior q_m and prior π_m distributions from m^{th} generation step.

4 Generative Adversarial Query Network

The proposed Generative Adversarial Query Network (GAQN) builds on the GQN architecture by introducing two novel contributions. We will explain both of these in the following subsections. The corresponding experimental results are presented in Section 5. Figure 1(c) shows the overall architecture of our method.

4.1 Adversarial loss

As illustrated in Figure 1(c), the proposed GAQN consists of three components: an encoder network *Enc*, a decoder network *Dec*, and a discriminator network *D*. The GAQN utilities the same *Enc* and *Dec* architecture as standard GQN to generate a novel view $x' = Dec(Enc(x, p), p_q)$. However, inspired by the recent advancement of GANs, we include an additional discriminator network D to distinguish between the generated fake images from the GQN and the ground truth view in the training data. In this way, the discriminator acts as a learn-able loss function that boosts the learning process of GQN.

Numerous works [1,21,24,25] have shown that the training of the GAN may be unstable due to the vanishing gradients problem caused by the binary cross entropy loss as defined in Equation (1) and (2). In this paper, we avoid the above problem by adopting the least-square loss function from the previously proposed Least Squares Generative Adversarial Networks (LSGANs) [20]. The idea of LSGANs has been proved to be effective since it tries to pull the fake samples closer to the decision boundary of the least-square loss function. Based on the distance between the sampled data and the decision boundary, LSGANs manages to generate better gradients to update its generator. Furthermore, LSGANs also proves to exhibit less mode-seeking behaviour [3] which also stabilize the training process. Equation (4) and (5) provide the generator and discriminator loss of LSGANs that we use to train the GQN decoder and our proposed discriminator, respectively.

$$\mathcal{L}_G^{LSGAN} = \mathbb{E}_{\boldsymbol{z}\sim P_{\boldsymbol{z}}}\Big[(D(G(\boldsymbol{z}))-1)^2\Big] \tag{4}$$

$$\mathcal{L}_D^{LSGAN} = \mathbb{E}_{\boldsymbol{z}\sim P_{\boldsymbol{z}}}\Big[(D(G(\boldsymbol{z})))^2\Big] + \mathbb{E}_{\boldsymbol{x}\sim P_{\boldsymbol{x}}}\Big[(D(x')-1)^2\Big] \tag{5}$$

Inspired by recent works [33,22] on GANs, our proposed Discriminator D network adopts the residual blocks architecture [13]. Instead of directly classifying the generated image as real or fake, we follow the patch-based discriminator [15] to restrict the attention to the structure in the local patches. Table 1 shows the design of our discriminator network.

The original GQN includes per-pixel variance annealing technique [7] that is aimed to focus the learning for the scene environment (wall, floor and sky), location and color of objects at the early stage of the training and enhance the low level details later. Even though the annealing strategy was shown to work, we argue that the adversarial loss is able to further speed up the learning and to generate sharper object details at the early stages of the training. Therefore, our GAQN can achieve better visualized novel view earlier than GQN.

Figure 2 illustrates the predicted novel view using the plain GQN architecture and the proposed GAQN architecture. Although the obtained GQN model successfully predicts the correct location and color of three objects in the 3D scene, their edges are blurry. Meanwhile our proposed method manages to produce sharper object edges, especially, to the middle green icosahedron. Further results and implementation details are provided in Section 5.

4.2 Feature-matching loss

Inspired by the recent works [2,25,30] on improving the stability of the GAN training, we add an extra feature matching loss to train the generator network.

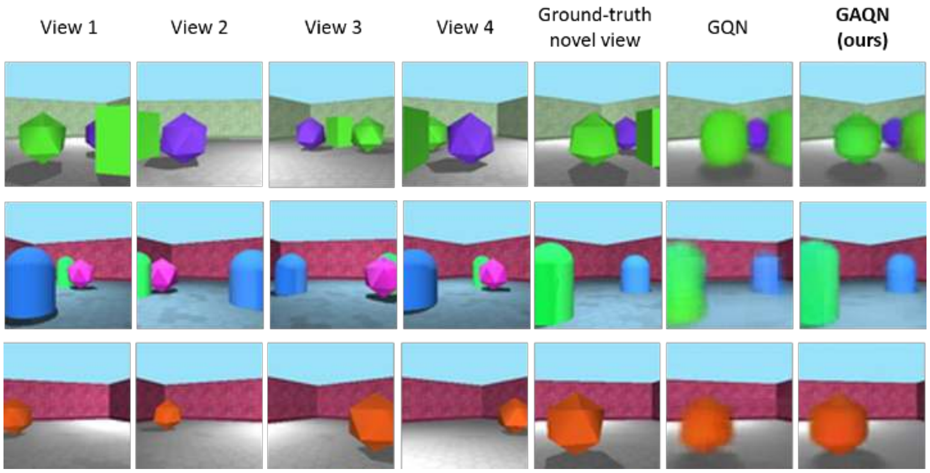

Fig. 2. Comparison between generated novel view using the GQN and the proposed GAQN architectures. The first four columns depict the training views of the scene and the fifth column shows the ground truth image for the novel view. The last two columns illustrate the results obtained using GQN architecture and the proposed approach, respectively. Our method is able to obtain clearly sharper images compared to the plain GQN architecture.

Table 1. Architecture of our discriminator D network using 3 residual blocks [13].

Layers	**Input size**	**Output size**
conv2 × 2, stride = 2, channels = 32, ReLU	$64 \times 64 \times 3$	$32 \times 32 \times 32$
ResBlock down 64	$32 \times 32 \times 32$	$16 \times 16 \times 64$
ResBlock down 128	$16 \times 16 \times 64$	$8 \times 8 \times 128$
ResBlock down 256	$8 \times 8 \times 128$	$4 \times 4 \times 256$
ResBlock down 512	$4 \times 4 \times 256$	$2 \times 2 \times 512$
conv1 × 1, stride = 1, channels = 1024, ReLU	$2 \times 2 \times 512$	$2 \times 2 \times 1024$

The main idea of this feature matching loss is to use the discriminator network as a feature extractor and guide the generator to generate data that matches the feature statistics of the real data. There are several approaches on exploiting the feature matching loss in training the generator.

Specifically, the common GAN generator loss as shown in Equation (1) is being replaced by a mean feature matching loss. It has been argued that this mean feature matching loss helps preventing the gradient vanishing problem during the training. In our research, we have already adopted the least square loss to address the above problem (Section 4.1) but there is no guarantee that the problem is completely solved. Therefore, we train our GAQN generator network using a unified loss function $\mathcal{L}_G^{GAQN}$ as the combination of LSGANs generator loss $\mathcal{L}_G^{LSGAN}$ and mean feature matching loss $\mathcal{L}_{FM}$. Let $f_D()$ be the mean of the output feature maps from the 3^{rd} layer (ResBlock down 128 in Table 1) of the discriminator network, the mean feature matching loss is define as follow:

$$\mathcal{L}_{FM} = ||\mathbb{E}_{\boldsymbol{x} \sim P_{\boldsymbol{x}}} f_D(x_{gt}) - \mathbb{E}_{\boldsymbol{z} \sim P_{\boldsymbol{z}}} f_D(x')||_2^2 \tag{6}$$

In this paper, we jointly train GAQN *Enc* and G by using $\mathcal{L}_{EG}^{GAQN}$ which is the conventional GQN ELBO loss (Equation (3)). The D parameters are being updated by the least squares loss (Equation (5)) from LSGANs, and we adopt the mean feature matching loss (Equation (6)) to update G. Finally, Equations (7), (8) and (9) show all the loss functions we have used to train our GAQN model.

$$\mathcal{L}_{G}^{GAQN} = \mathcal{L}_{G}^{LSGAN} + \mathcal{L}_{FM} \tag{7}$$

$$\mathcal{L}_{D}^{GAQN} = \mathcal{L}_{D}^{LSGAN} \tag{8}$$

$$\mathcal{L}_{EG}^{GAQN} = \mathcal{L}_{GQN} \tag{9}$$

5 Experiments

5.1 Experimental settings

We evaluate our method using the *rooms ring camera* dataset that was provided and used by Eslami et al. in [7]. The dataset contains various rendered 3D square rooms that contain random objects of various shapes, colors and locations. Moreover, the scene textures, walls and lights are also randomly generated. Therefore, the task of predicting the novel views on this data-set is a relatively challenging problem.

The original GQN construction proposed in [7] consumes a large amount of computing and memory resources. Due to computational restrictions, we demonstrate the advantages of the proposed architectural changes using a smaller version of the basic GQN network and a subset of the training data. Originally, training a 12 generative-layers GQN model with batch size of 36 requires 4 NVIDIA K80 GPUs as shown in [7]. In this paper, we use a single NVIDIA Tesla P100 GPU to train our GAQN model which has 8 generative layers and batch size is 20. As far as we experimented, this change would not affect the quality of results. Despite of the smaller network size, our method is able to produce results (see figures 2 and 4) that are very close to those presented in the original work [7] that uses clearly larger network and training set. This further, emphasized the benefits of the proposed architectural modifications. Moreover, we only use the first halve of the original GQN's training and testing data for faster training procedure.

We implement our model on PyTorch [23] and our GAQN model is trained end-to-end using ADAM optimization [18] with hyper-parameter $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The generator network is trained using the learning rate of 10^{-4} and the discriminator is trained using a learning rate of 4×10^{-4}. Recent work by Heusel et al. [14] shows that if we update the generator slower than the discriminator then it helps to reach the convergence easier.

5.2 Results

Table 2. Comparison of training, testing loss, KL testing loss and SSIM testing score between our GAQN model (GQN + LSGANs + FM), original GQN and a variant (training GQN with LSGANs loss).

	GQN	GQN + LSGANs	**GAQN (ours)**
Training loss	7012	6988	**6951**
Testing loss	7003	6974	**6957**
KL testing loss	17.59	19.36	**22.91**
SSIM	0.742	0.815	**0.865**

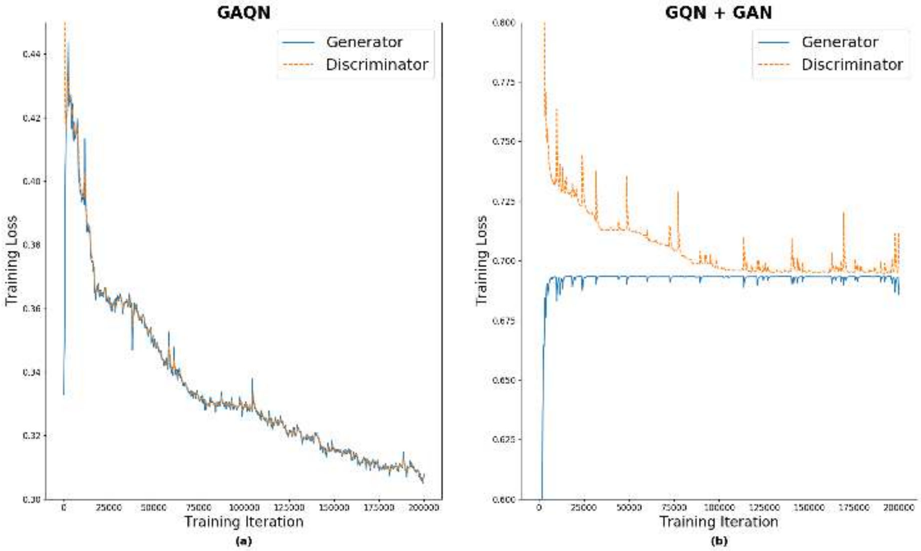

Fig. 3. Comparison of generator and discriminator training loss between our proposed GAQN (a) and GQN + GAN (b). Both generator and discriminator of GQN+GAN are suffering from mode collapsing and vanishing gradients. Using the least-square loss and the mean feature loss, our GAQN achieves a stable learning process.

Our approach relies on two new components, namely, an adversarial training pipeline using the least-square GANs loss and the discriminator feature matching loss functions to enhance the previously proposed GQN model. In this section, we will experimentally investigate the impact of the both components. The results corresponding to only GQN and adversarial loss are denoted as (GQN + LSGANs), whereas the results with the proposed full model are denoted as

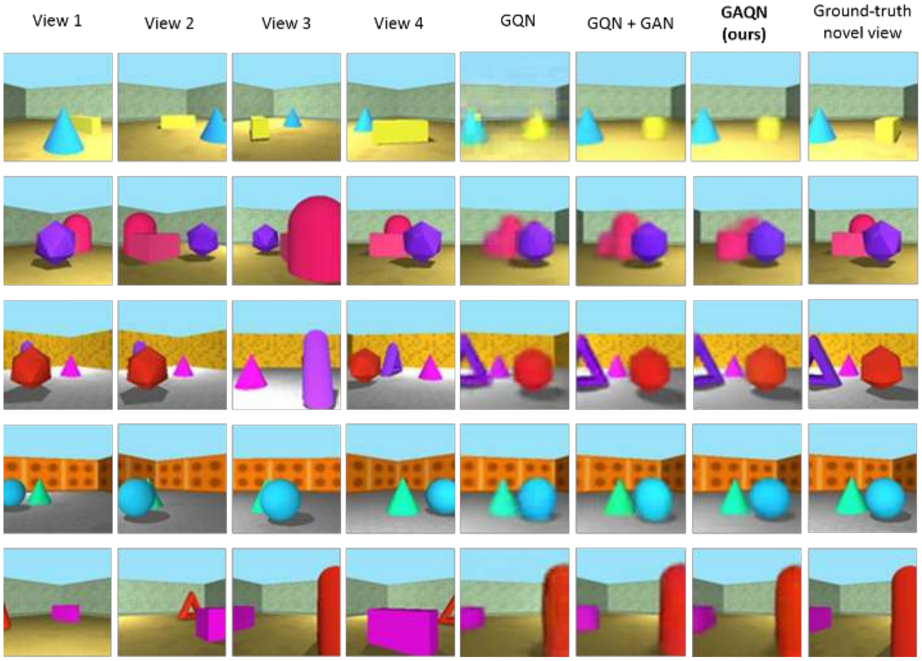

Fig. 4. Generated novel view comparison between our proposed GAQN and variants

GAQN. We train the above methods using the same hyper-parameters (reported in Section 5.1) and record their training and testing loss for evaluation purposes. We also use the structure similarity index (SSIM) to assess on the quality of the target image and the predicted novel on a held-out test set.

Table 2 contains the obtained results. The proposed full GAQN method has clearly the smallest training and testing loss and the largest Kullback–Leibler (KL) testing loss. The KL loss represents the distance between the estimated and true distribution produced by GQN's generation network and this KL loss is the second term in the ELBO loss (Equation (3)). If the model manages to produce high KL loss then the predicted novel view tends to be close to the ground-truth data. It is also evident that the adversarial loss is able to improve the results compared to the plain GQN architecture. However, the largest gain is obtained by combining both of the proposed contributions.

In Figure 4, we show qualitative examples of the generated novel views produced by different versions of the proposed GAQN method and the baseline. Although the baseline manages to correctly generate the colors and positions of objects in most cases, theirs edges are blurry. Based on the SSIM, our GAQN model achieves highest score and produce sharper edges on predicted images.

Finally, we illustrate how the least squares loss of LSGANs and discriminator feature matching loss affect the training procedure of the method. In this experiment, we compare the generator and discriminator training loss of our proposed

GAQN and GQN + GAN. As can be seen in Figure 3 (b), the training procedure of GQN + GAN is highly unstable due to mode collapsing and diminishing gradients. Our GAQN model eliminates both problems by using the least-squares loss and the discriminator mean feature loss as shown in Figure 3 (a).

6 Conclusion

We have introduced an novel adversarial training pipeline to improve the previously proposed GQN network architecture. Our experimental results demonstrate that training an additional discriminator network encourages the GQN model to predict more accurate novel views. Moreover, using the combination of least square loss and the feature matching loss helps stabilizing both generator and discriminator training process. In our future work, we will explore how to generate novel views in the bigger scale of indoor scenes where there are more objects and different lighting conditions.

References

1. Arjovsky, M., Chintala, S., Bottou, L.: Wasserstein generative adversarial networks. In: Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017. pp. 214–223 (2017), http://proceedings.mlr.press/v70/arjovsky17a.html
2. Bao, J., Chen, D., Wen, F., Li, H., Hua, G.: CVAE-GAN: fine-grained image generation through asymmetric training. In: IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017. pp. 2764–2773. IEEE Computer Society (2017). https://doi.org/10.1109/ICCV.2017.299, https://doi.org/10.1109/ICCV.2017.299
3. Bishop, C.M.: Pattern Recognition and Machine Learning (Information Science and Statistics). Springer-Verlag, Berlin, Heidelberg (2006)
4. Chaurasia, G., Duchene, S., Sorkine-Hornung, O., Drettakis, G.: Depth synthesis and local warps for plausible image-based navigation. ACM Trans. Graph. **32**(3), 30:1–30:12 (Jul 2013). https://doi.org/10.1145/2487228.2487238, http://doi.acm.org/10.1145/2487228.2487238
5. Chen, S.E., Williams, L.: View interpolation for image synthesis. In: Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques. pp. 279–288. SIGGRAPH '93, ACM, New York, NY, USA (1993). https://doi.org/10.1145/166117.166153, http://doi.acm.org/10.1145/166117.166153
6. Dosovitskiy, A., Springenberg, J.T., Brox, T.: Learning to generate chairs with convolutional neural networks. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1538–1546 (June 2015). https://doi.org/10.1109/CVPR.2015.7298761
7. Eslami, S.M.A., Jimenez Rezende, D., Besse, F., Viola, F., Morcos, A.S., Garnelo, M., Ruderman, A., Rusu, A.A., Danihelka, I., Gregor, K., Reichert, D.P., Buesing, L., Weber, T., Vinyals, O., Rosenbaum, D., Rabinowitz, N., King, H., Hillier, C., Botvinick, M., Wierstra, D., Kavukcuoglu, K., Hassabis, D.: Neural scene representation and rendering. Science **360**(6394), 1204–1210

(2018). https://doi.org/10.1126/science.aar6170, http://science.sciencemag.org/content/360/6394/1204

8. Flynn, J., Neulander, I., Philbin, J., Snavely, N.: Deepstereo: Learning to predict new views from the world's imagery. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
9. Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2. pp. 2672–2680. NIPS'14, MIT Press, Cambridge, MA, USA (2014), http://dl.acm.org/citation.cfm?id=2969033.2969125
10. Gregor, K., Besse, F., Jimenez Rezende, D., Danihelka, I., Wierstra, D.: Towards conceptual compression. In: Lee, D.D., Sugiyama, M., Luxburg, U.V., Guyon, I., Garnett, R. (eds.) Advances in Neural Information Processing Systems 29, pp. 3549–3557. Curran Associates, Inc. (2016), http://papers.nips.cc/paper/6542-towards-conceptual-compression.pdf
11. Gregor, K., Danihelka, I., Graves, A., Rezende, D., Wierstra, D.: Draw: A recurrent neural network for image generation. In: Bach, F., Blei, D. (eds.) Proceedings of the 32nd International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 37, pp. 1462–1471. PMLR, Lille, France (07–09 Jul 2015), http://proceedings.mlr.press/v37/gregor15.html
12. Habtegebrial, T., Varanasi, K., Bailer, C., Stricker, D.: Fast view synthesis with deep stereo vision. arXiv preprint arXiv:1804.09690 (2018)
13. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016. pp. 770–778. IEEE Computer Society (2016). https://doi.org/10.1109/CVPR.2016.90, https://doi.org/10.1109/CVPR.2016.90
14. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) Advances in Neural Information Processing Systems 30, pp. 6626–6637. Curran Associates, Inc. (2017), http://papers.nips.cc/paper/7240-gans-trained-by-a-two-time-scale-update-rule-converge-to-a-local-nash-equilibrium.pdf
15. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: IEEE Conference on Computer Vision and Pattern Recognition (2017)
16. Jimenez Rezende, D., Eslami, S.M.A., Mohamed, S., Battaglia, P., Jaderberg, M., Heess, N.: Unsupervised learning of 3d structure from images. In: Lee, D.D., Sugiyama, M., Luxburg, U.V., Guyon, I., Garnett, R. (eds.) Advances in Neural Information Processing Systems 29, pp. 4996–5004. Curran Associates, Inc. (2016)
17. Karras, T., Aila, T., Laine, S., Lehtinen, J.: Progressive growing of GANs for improved quality, stability, and variation. In: International Conference on Learning Representations (2018), https://openreview.net/forum?id=Hk99zCeAb
18. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
19. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013)
20. Mao, X., Li, Q., Xie, H., Lau, R.Y.K., Wang, Z., Smolley, S.P.: On the effectiveness of least squares generative adversarial networks. IEEE transactions on pattern analysis and machine intelligence (2018)

21. Metz, L., Poole, B., Pfau, D., Sohl-Dickstein, J.: Unrolled generative adversarial networks. arXiv preprint arXiv:1611.02163 (2016)
22. Miyato, T., Kataoka, T., Koyama, M., Yoshida, Y.: Spectral normalization for generative adversarial networks. In: International Conference on Learning Representations (2018), https://openreview.net/forum?id=B1QRgziT-
23. Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A.: Automatic differentiation in pytorch. In: NIPS-W (2017)
24. Qi, G.J.: Loss-sensitive generative adversarial networks on lipschitz densities. arXiv preprint arXiv:1701.06264 (2017)
25. Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training gans. In: Advances in neural information processing systems. pp. 2234–2242 (2016)
26. Scharstein, D.: View Synthesis Using Stereo Vision. Springer-Verlag, Berlin, Heidelberg (1999)
27. Seitz, S.M., Dyer, C.R.: Physically-valid view synthesis by image interpolation. In: Proceedings IEEE Workshop on Representation of Visual Scenes (In Conjunction with ICCV'95). pp. 18–25 (June 1995). https://doi.org/10.1109/WVRS.1995.476848
28. Sun, S.H., Huh, M., Liao, Y.H., Zhang, N., Lim, J.J.: Multi-view to novel view: Synthesizing novel views with self-learned confidence. In: The European Conference on Computer Vision (ECCV) (September 2018)
29. Tatarchenko, M., Dosovitskiy, A., Brox, T.: Multi-view 3d models from single images with a convolutional network. In: European Conference on Computer Vision (ECCV) (2016)
30. Warde-Farley, D., Bengio, Y.: Improving generative adversarial networks with denoising feature matching. In: International Conference on Learning Representations 2017 (Conference Track) (2017), https://openreview.net/forum?id=S1X7nhsxl
31. Wu, J., Zhang, C., Xue, T., Freeman, B., Tenenbaum, J.: Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. In: Lee, D.D., Sugiyama, M., Luxburg, U.V., Guyon, I., Garnett, R. (eds.) Advances in Neural Information Processing Systems 29, pp. 82–90. Curran Associates, Inc. (2016)
32. Yan, X., Yang, J., Yumer, E., Guo, Y., Lee, H.: Perspective transformer nets: Learning single-view 3d object reconstruction without 3d supervision. In: Lee, D.D., Sugiyama, M., Luxburg, U.V., Guyon, I., Garnett, R. (eds.) Advances in Neural Information Processing Systems 29, pp. 1696–1704. Curran Associates, Inc. (2016), http://papers.nips.cc/paper/6206-perspective-transformer-nets-learning-single-view-3d-object-reconstruction-without-3d-superv
pdf
33. Zhang, H., Goodfellow, I., Metaxas, D., Odena, A.: Self-attention generative adversarial networks. arXiv preprint arXiv:1805.08318 (2018)
34. Zhang, Y., Xu, W., Tong, Y., Zhou, K.: Online structure analysis for real-time indoor scene reconstruction. ACM Trans. Graph. **34**(5), 159:1–159:13 (Nov 2015). https://doi.org/10.1145/2768821, http://doi.acm.org/10.1145/2768821
35. Zhou, T., Tucker, R., Flynn, J., Fyffe, G., Snavely, N.: Stereo magnification: Learning view synthesis using multiplane images. In: SIGGRAPH (2018)
36. Zhou, T., Tulsiani, S., Sun, W., Malik, J., Efros, A.A.: View synthesis by appearance flow. In: European conference on computer vision. pp. 286–301. Springer (2016)