Multi-view 3D Models from Single Images with a Convolutional Network: Supplementary Material

Maxim Tatarchenko, Alexey Dosovitskiy, Thomas Brox

Department of Computer Science
University of Freiburg
{tatarchm, dosovits, brox}@cs.uni-freiburg.de

We present experimental results showing the effect of realistic rendering and the effect of adversarial training. We also analyze how the internal representation changes when the network is presented with different input views of the same object.

1 Realistic rendering

As mentioned in the paper, we found that in order to achieve better generalization to real images special care has to be taken when rendering the training data. We trained networks with two kinds of training data: "realistic" and "basic".

The "realistic" rendering is described in the main paper: we randomly sampled the number of light sources, their intensities and the locations; performed alpha compositioning to avoid sharp transition between the model and the background; and additionally smoothed the car image with a Gaussian filter. The "basic" rendering is with two light sources of fixed intensity, without alpha compositioning and smoothing.

Figure 1 compares the results of networks trained on these two kinds of data. The network trained on "basic" data (bottom row for each model) fails to correctly estimate the car shape in all cases but one. The network trained with "realistic" data performs much better, demonstrating how the quality of the training data is crucial for generalization to real images.

2 Adversarial training

Tasks involving image generation are still mostly solved by optimizing L_2 objective, which is robust but often leads to blurred results. This happens because of the fundamental uncertainty associated with novel view estimation, which in case of Euclidean loss leads to predicting the average of all possibilities. Alternatively, one can use the idea of adversarial training introduced by Goodfellow et al. [1]. The aim is to train a *generator* G_ψ (parametrized by a neural network with weights ψ) which takes random noise as input and generates realistic images. This is achieved by training the generator concurrently with another neural network – a *discriminator* D_φ. The discriminator aims to distinguish the generated images from real ones, while the generator aims to trick the discriminator. Mathematically, the parameters φ of the discriminator are trained

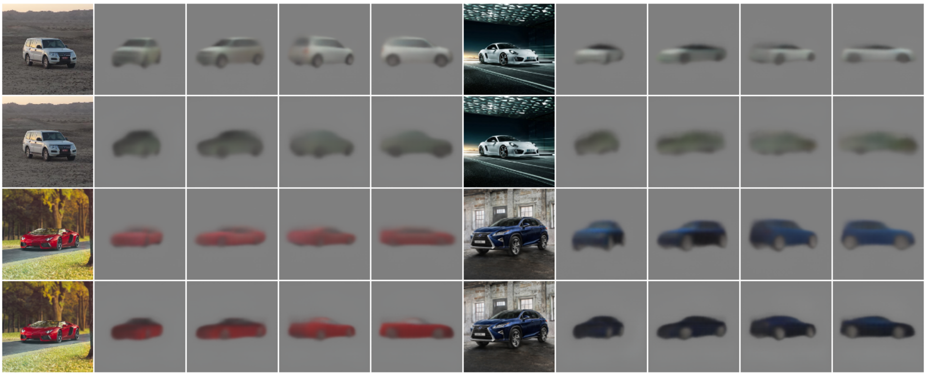

Fig. 1. Predictions from the network trained on ”realistic” data (top for each model) compared with those from the network trained on ”basic” data (bottom for each model).

by minimizing

$$\mathcal{L}_{discr} = -\sum_i \log(D_\varphi(\mathbf{y}_i)) + \log(1 - D_\varphi(G_\psi(\mathbf{x}_i))), \tag{1}$$

where $\mathbf{x}_i$ is the noise sample and $\mathbf{y}_i$ is the target sample from the training set. The generator is trained to minimize

$$\mathcal{L}_{adv} = -\sum_i \log D_\varphi(G_\psi(\mathbf{x}_i)). \tag{2}$$

Conditional GANs were successfully applied to future prediction in videos [2] and other image generation tasks [3] and demonstrated superior performance over standard squared Euclidean objective. This motivated us to use adversarial training to decrease blur in car images predicted by the network. Namely, we minimized

$$\mathcal{L}_{gen} = \mathcal{L}_{euc} + \alpha\mathcal{L}_{adv}, \tag{3}$$

where $\mathcal{L}_{adv}$ was trained as described above, with the difference that our generator was conditioned on the input image instead of noise. In our experiments we used $\alpha = 0.01$. We used the same generator as for all other experiments. The discriminator is a convolutional network identical to the encoder of the generator. It takes both input and output view as input.

Comparison of viewpoint prediction results with and without adversarial loss is shown in Figure 2. While adversarial training does lead to sharper predictions, this happens at the cost of increased image noise and worse estimate of the car shape. Moreover, the network with adversarial loss is much more sensitive to hyperparameter settings. We therefore concentrated on getting best results with standard non-adversarial losses. Still, we believe adversarial training could be useful to increase the visual quality of the network predictions and see it as an interesting direction of future research.

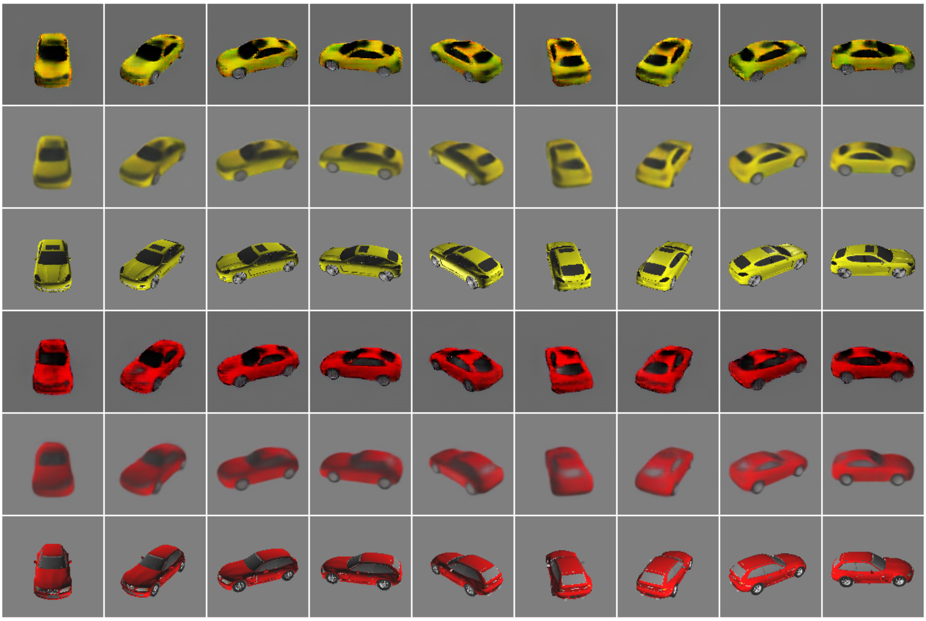

Fig. 2. Predictions with networks trained with adversarial and squared Euclidean loss. For each model, top row: with adversarial loss, second row: without adversarial loss, bottom row: ground truth.

3 Intermediate representation

We studied the properties of the internal representation by computing it for 3 different views of 5 different car models. Figure 3 shows the input data and the matrix of pairwise Euclidean distance between the cars in the hidden space. 3×3 diagonal blocks indicate that different input views of the same car lead to a similar hidden representation. The representation of the second car is quite close to that of the fifth one (off-diagonal blue elements in the matrix) because both cars have similar shape.

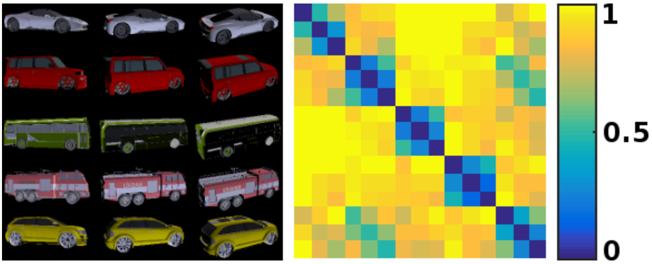

Fig. 3. Pairwise distances between the hidden vectors of five cars and three input views. Different input views of the same model lead to similar hidden representations.

References

1. Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A.C., Bengio, Y.: Generative adversarial nets. In: Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada. (2014) 2672–2680
2. Mathieu, M., Couprie, C., LeCun, Y.: Deep multi-scale video prediction beyond mean square error. CoRR **abs/1511.05440** (2015)
3. Dosovitskiy, A., Brox, T.: Generating images with perceptual similarity metrics based on deep networks. CoRR **abs/1602.02644** (2016)