Supplementary Material for "Adversarial Inverse Graphics Networks: Learning 2D-to-3D Lifting and Image-to-Image Translation from Unpaired Supervision"

Hsiao-Yu Fish Tung * Adam W. Harley * William Seto * Katerina Fragkiadaki
Carnegie Mellon University
{htung,aharley,wseto,katef}@cs.cmu.edu

1. Parametric vs. non-parametric decoders

Here we discuss the benefits of using non-parametric and domain-specific renderers, over learned decoders. Both the proposed model and CycleGAN [5] can be viewed as autoencoders: the input is first transformed into a target domain, and then transformed back to its original space. A parametric decoder could be more desirable, for the reason that we do not need to hand-engineer a mapping function from the target domain back to the inputs. However, simply using reconstruction loss and adversarial loss does not guarantee that the predictions look spatially similar to the inputs. In tasks such as image-to-image translation, spatial precision can be of critical importance. With a parametric decoder, the transformed input can be viewed as a information bottleneck, and as long as the decoder can correctly "guess" the final output from the transformed input (*i.e.*, the code), the code is valid and the solution is optimal.

To support this point, we conduct an experiment on image inpainting using the MNIST dataset. Similar to the parametric encoder-decoder described in the main text, the network has two main parts: (1) an encoder that transforms the input (a partially obscured image of a digit) into prediction (a hallucinated digit), and (2) a decoder that transforms the prediction back into the input. Instead of using convolutional layers, which have an architectural bias on preserving spatial relationships, we use fully-connected layers in both the encoder and the decoder. This is important, because such architectural conveniences are unavailable in less-structured tasks, such as 3D pose prediction and SfM. We train the model with a reconstruction loss on the decoder, and adversarial loss on the encoder.

The results are shown in Figure 1. While inpainting, the encoder (incorrectly) transforms many of the digits into other digits. For instance, several obscured "1" images are inpainted as "4". In the parametric decoding process, however, these errors are *undone*, and the original input is recovered successfully. In other words, the decoder takes the burden of the reconstruction loss, allowing the encoder to learn an inaccurate latent space. Parameter-free rendering avoids this problem.

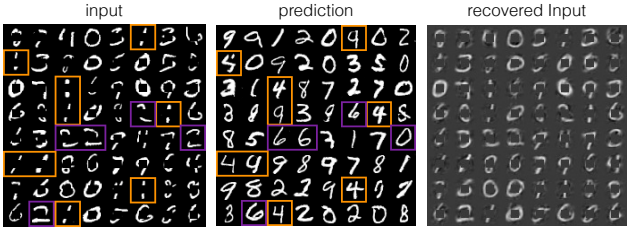


Figure 1. Digit inpainting using an encoder-decoder architecture with fully-connected layers. Many predictions are incorrect, while the recovered inputs are accurate. Orange squares highlight instances of the digit "1" transformed into other digits; purple squares highlight instances of the digit "2" transformed into other digits.

2. Additional experiments and details

In the sections to follow, we provide implementation details, including architecture descriptions for the generator and discriminator in each task, and training details. Additionally, we provide more experimental results.

2.1. 3D human pose estimation from static images

Figure 2.1 shows the architecture of our generator network for 3D human pose estimation from a single RGB image. Our generator predicts weights over the shape bases α, rotation R, translation T and focal length f, as described in our paper. The generator takes as input a set of 2D body joint heatmaps. We use convolutional pose machines [3] to estimate 2D keypoints, and convert them into heatmaps by creating a Gaussian distribution centered around each 2D keypoint. The network consists of 8 convolutional layers with leaky ReLU activations and batch normalization and two fully connected layers at the end that map to the de-

*equal contribution

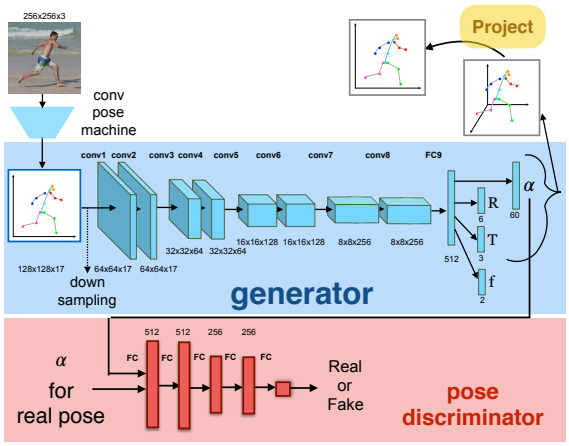


Figure 2. **Generators and discriminators' architectures for the task of 3D human pose estimation from a single image.**

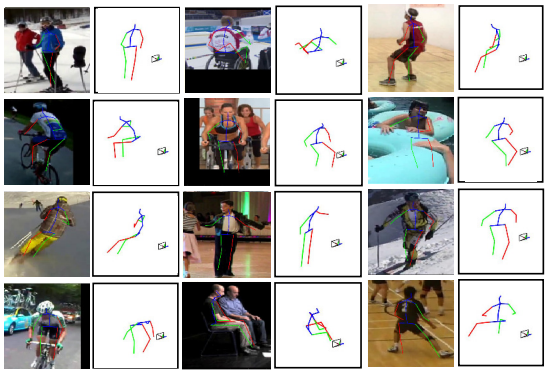


Figure 3. **Predicted 3D human poses in MPII dataset using the supplied ground-truth 2D keypoints as input.**

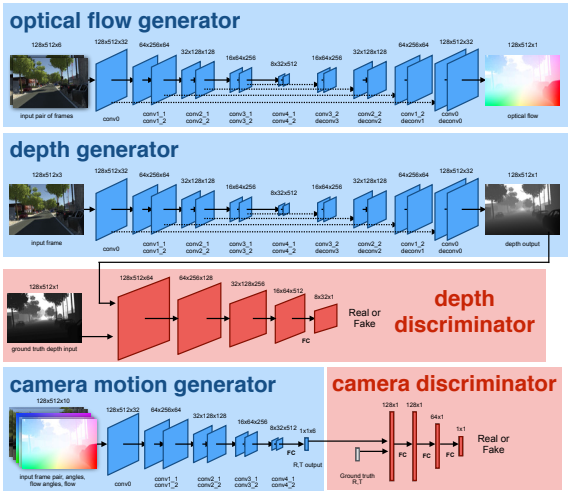


Figure 4. **Generator and discriminator architectures for Structure from Motion.** Dashed lines indicate skip connections.

sired outputs. The width, height and number of channels for each layer is specified in Figure 6. The discriminator for this task consists of five fully connected layers with featuremap depth 512, 512, 256, 256 and 1, with a leaky ReLU and batch normalization after each layer. The discriminator takes all values output from the generator (*i.e.*, α, R, T, f) as input.

In all experiments, we set the variance for the Gaussian heatmap σ to 0.25, and the dimensionality of our PCA shape basis to 60 (out of 96 total bases). The dimensionality reduction is small, and indeed, we only use basis weights for ease of prediction, relying on our adversarial priors (rather than PCA) to regularize the 3D shape prediction. We use gradient descent for both generator and discriminator training. Learning rate for reconstruction loss is set to 0.00001 and learning rate for the adversarial loss is set to 0.0001. All parameters are initialized with random sampling from zero mean normal distributions with variance of 0.02.

In Figure 3, we show predicted 3D human poses on images from the MPII dataset [1] using the ground-truth 2D keypoints available. Our model generalizes well *on unseen images without any further self-supervised finetuning*, though we would expect additional self-supervised finetuning to further improve performance.

2.2. Structure from Motion

Our generator networks for the task of structure from motion is illustrated in Figure 2.1. It includes three encoder-decoder convolutional networks with skip connections, which solve for optical flow, depth, and camera motion. The egomotion network uses RGB, optical flow and an angle field as input, and estimates the camera motion in SE(3). The depth network takes an RGB image as input and predicts logdepth. The depth discriminator consists of four convolution layers with batch normalization on the second and the third layers, and leaky ReLU activation after each layer. The depth discriminator is fully convolutional as we are interested in the realism of every depth patch, as opposed to the depthmap as a whole.

The egomotion discriminator is a 3-layer fully-connected network that takes $\{R, T\}$ matrices as input. The hidden layers have 128, 128, and 64 neurons, respectively, with batch normalization and a leaky ReLU after each layer.

Stabilizing training. In order to make sure that generators and discriminators progress together during training, we update the generator only when the discriminator has low enough loss. We add an updating heuristic such that if the likelihood loss of the discriminator is above a threshold θ_d, we do not update the generator. While discriminator is strong enough (below this threshold) and the generator is relatively weak (below a different threshold θ_g), we update the generator twice in the iteration. In the experiments, we set θ_d to 0.695 and θ_g to 0.75.

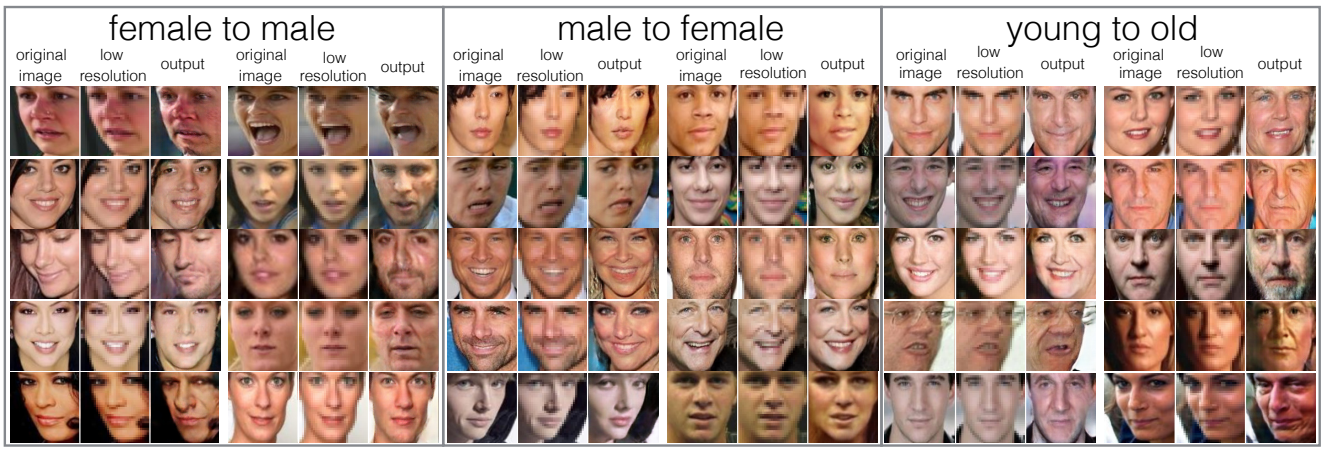
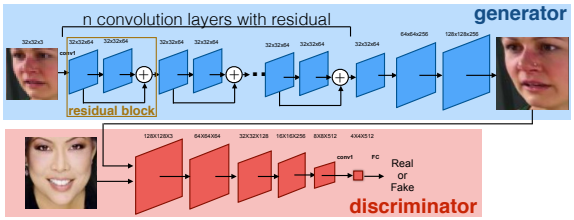
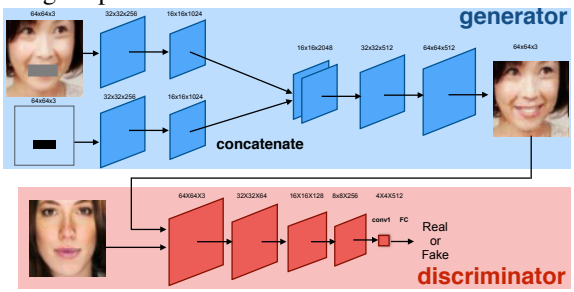


Figure 5. AIGN on **gender transformation (female to male, male to female)** and **age transformation (young to old).**



(a) Generator's and discriminator's architectures for image super-resolution.



(b) Generator and discriminator architectures for image inpainting.

Figure 6. **Architectures for AIGN.**

2.3. Image Super-Resolution

In Figure 6, we show the architecture of the generator and discriminator for image super-resolution. The input image is first passed through a convolutional layer with 64 channels, then n "residual blocks". Each residual block consists of two convolutional layers, with a batch normalization after each convolution layer and ReLU activation after the first batch normalization. The output from the last block is further passed to two deconvolution layers and generates the final image. The discriminator for this task consists of five convolution layers that use leaky ReLU activations and batch normalization, and one fully-connected layer that outputs one final value. In all experiments, we use Adam optimizer with learning rate 0.0001.All parameters are initialized with truncated normal distribution with variance 0.02.

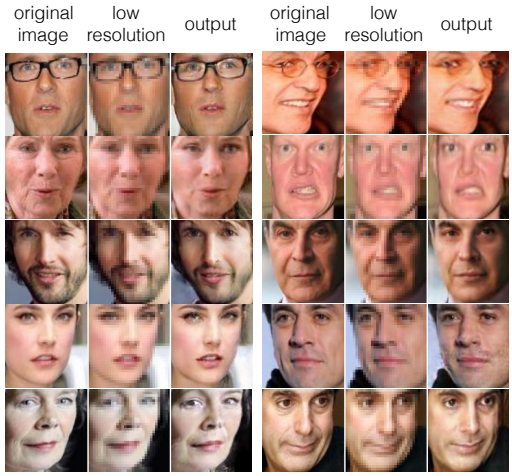


Figure 7. AIGN on **age transformation (old to young).**

In Figures 8, 9 and 10, we compare our model with Attribute2Image [4] and with Unsupervised Image Translation [2] for gender and age transformations. We use the code provided by the authors for our comparisons. In Figures 5 and 7, we show additional results of our model on gender and age transformation.

2.4. Inpainting

Figure 6 illustrates the architecture of our generator and discriminator for image inpainting. The occluded input image and the corresponding mask are separately passed through two convolution layers, and then concatenated. The concatenated outputs are then passed to three deconvolutional layers to generate the inpainted image. The discriminator consists of four convolutional layers with leaky ReLU and batch normalization layers, and one fully connected

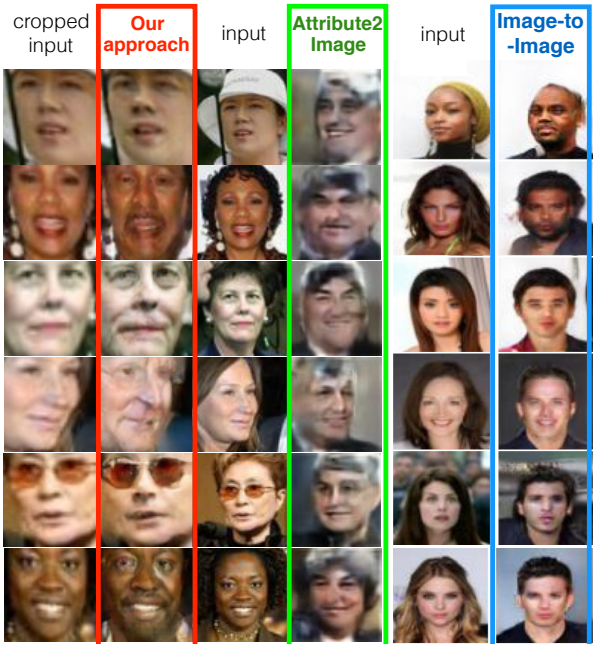


Figure 8. Comparison with Attribute2Image [4] and Unsupervised Image to Image Translation [2] on **Gender transformation (female to male)**. Input to our model is a tight crop around the face, tighter than the crop used by [4]. The proposed AIGN (*Column 2*) provides more realistic results that better preserves the "identity" of the subject while changing its gender, in comparison to previous work [4] (*Column 4*). We further show gender transformations from the model of [2] (*Columns 5,6*) where as we see the identity preservation is much weaker. Code is not available so we just paste some results from their paper.

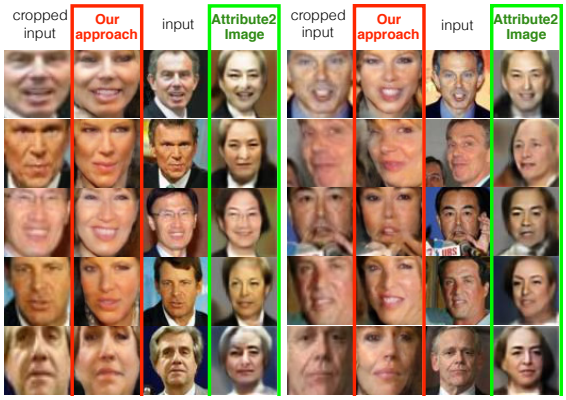


Figure 9. Comparison of AIGN with Attribute2Image [4] on **gender transformation (male to female)**.

layer that outputs one final value. In all experiments, we use the Adam optimizer, with a learning rate 1e−4All parameters are initialized from the truncated Normal distribution, with variance 0.02.

In Figure 11, we show additional results on biased inpainting for making bigger lips.

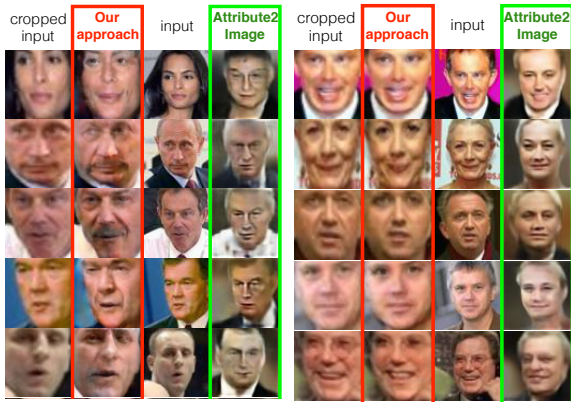


Figure 10. Comparison of AIGN with Attribute2Image [4] on **age transformation (left: young to old; right: old to young).**

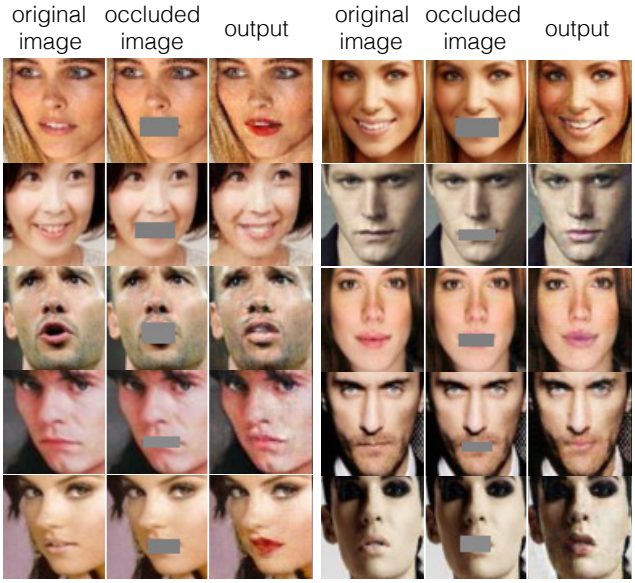


Figure 11. Additional results of AIGN on **biased image inpainting (big lips).**

References

[1] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[2] H. Dong, P. Neekhara, C. Wu, and Y. Guo. Unsupervised image-to-image translation with generative adversarial networks. *CoRR*, abs/1701.02676, 2017.

[3] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh. Convolutional pose machines. In *CVPR*, 2016.

[4] X. Yan, J. Yang, K. Sohn, and H. Lee. Attribute2image: Conditional image generation from visual attributes. *CoRR*, abs/1512.00570, 2015.

[5] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. *arXiv preprint arXiv:1703.10593*, 2017.