The GAN that Warped: Semantic Attribute Editing with Unpaired Data Supplement

1. High-resolution Flickr faces

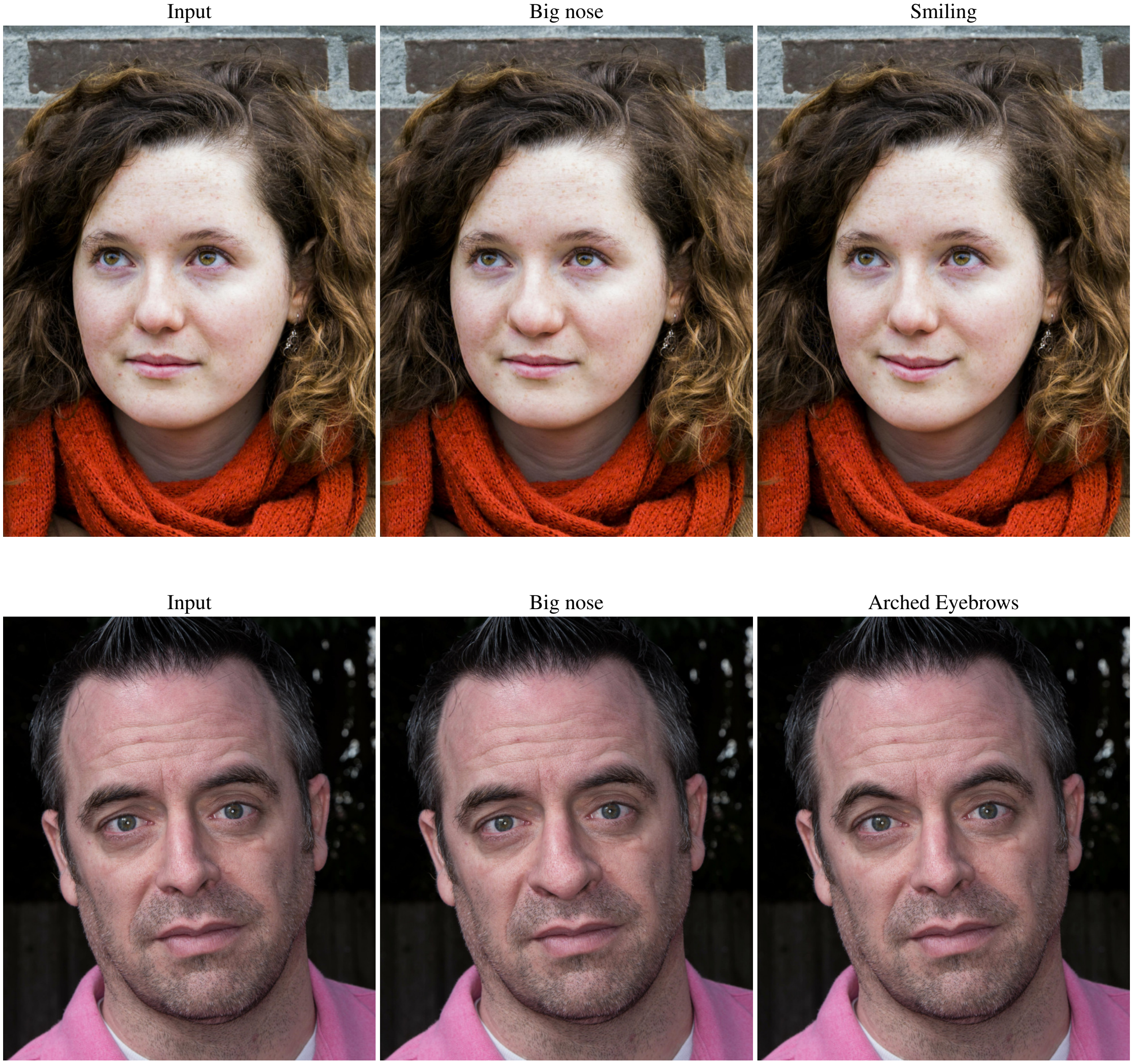

Figure 1: Additional results of our model on high-resolution images. Our model predicts warps at low resolution that can then be resized and applied to high resolution images. The model is able to keep the content and identity at high resolution. Please see supplemental videos demonstrating animated edits. Input images courtesy of Flickr users Kenneth DM and Randall Pugh. (Zoom in for details)

2. High-resolution Flickr birds

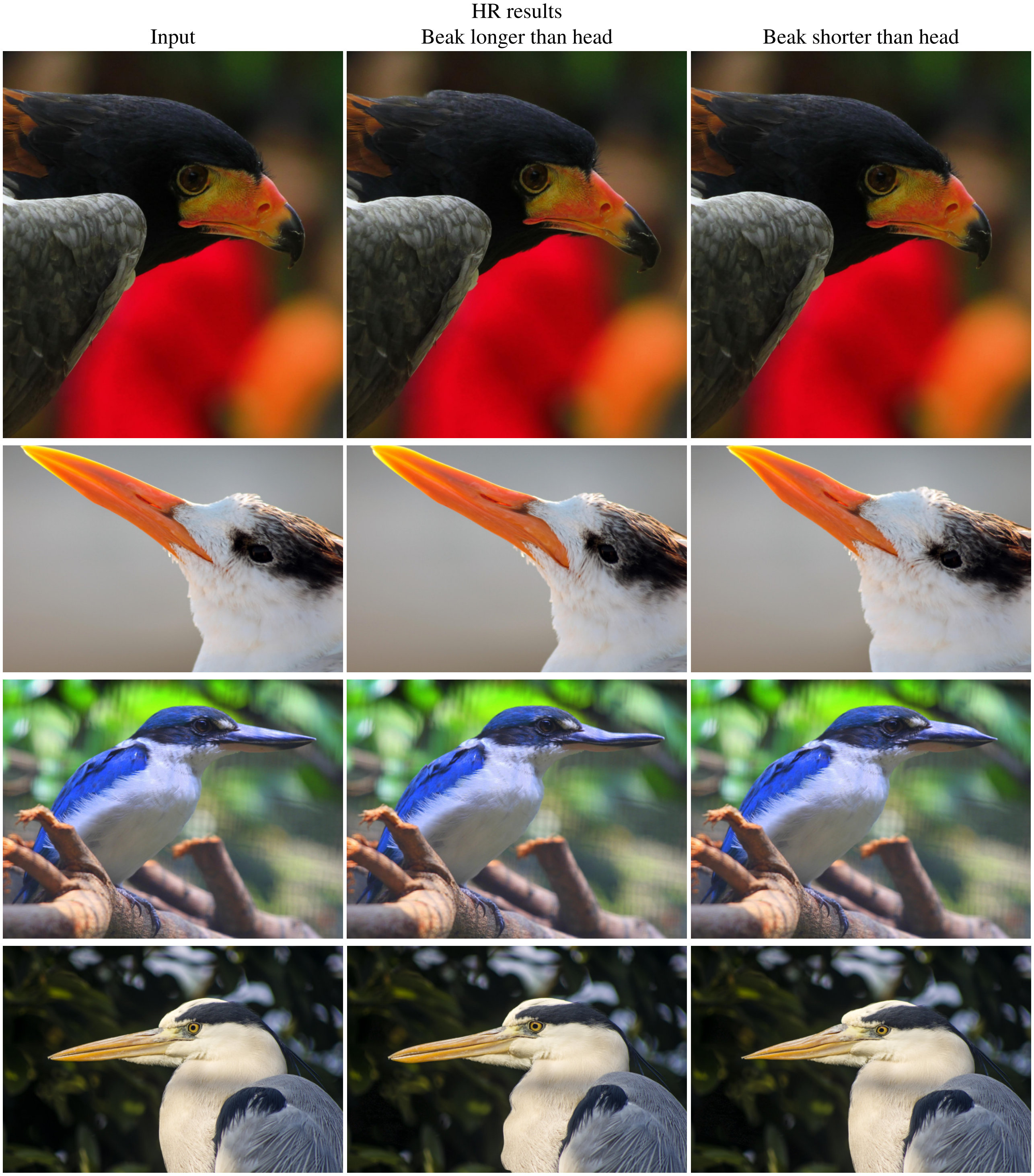

Figure 2: Additional results of our model on high-resolution images. Our model predicts warps at low resolution that can then be resized and applied to high resolution images. The model is able to keep the content and identity at high resolution. Please see supplemental videos demonstrating animated edits. Input images courtesy of Flickr users mickey, Lisa Leonardelli, and Andrey Grushnikov.

3. Qualitative results on CelebA

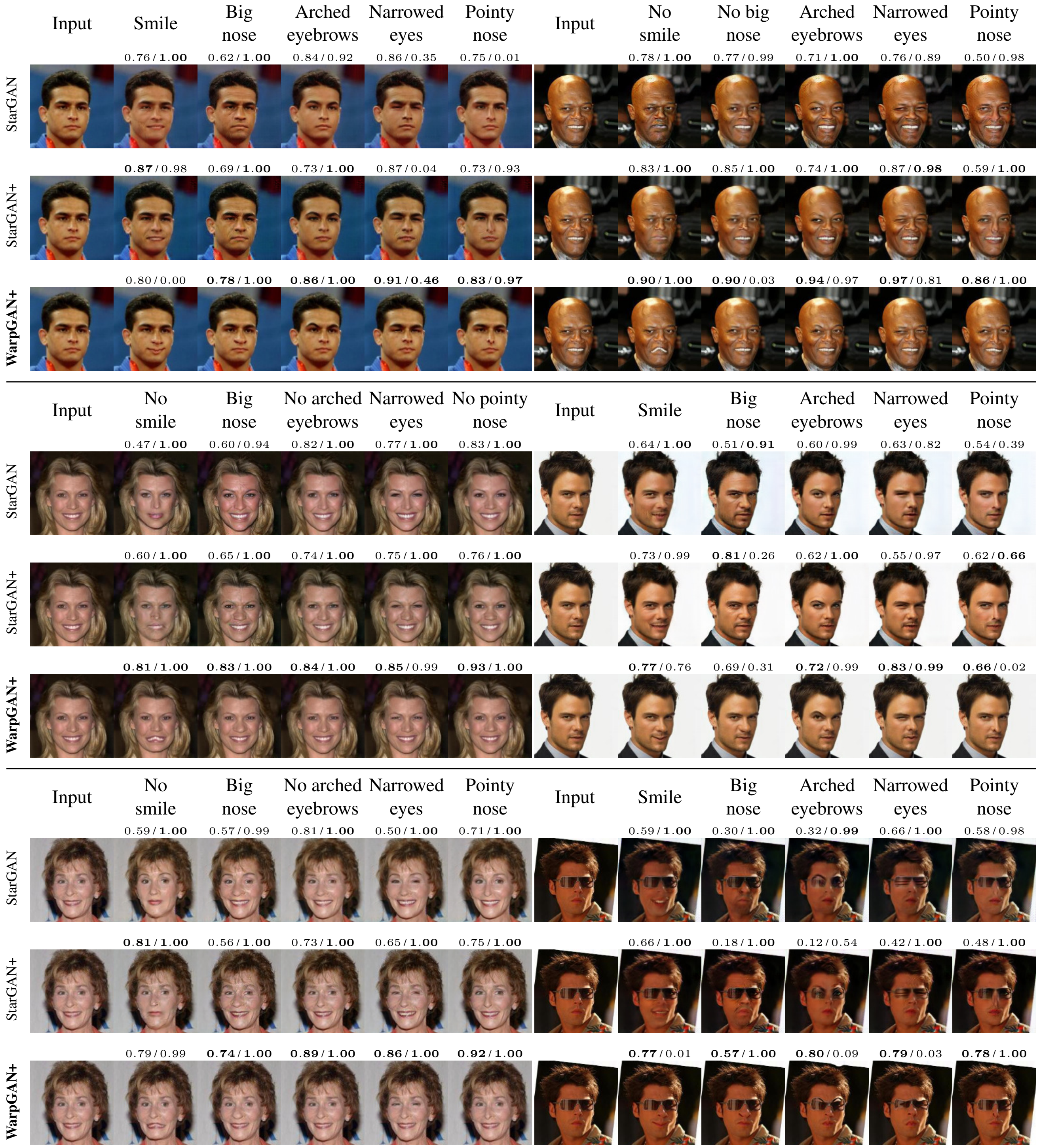

Figure 3: Comparison to previous work on the CelebA dataset. From a given input image, first column, each method attempts to transfer the semantic attribute in its corresponding column. On top of each image the re-identification score and the classification accuracy are shown as (id / cls) (higher is better). (Zoom in for details)

4. Qualitative results on Cub200

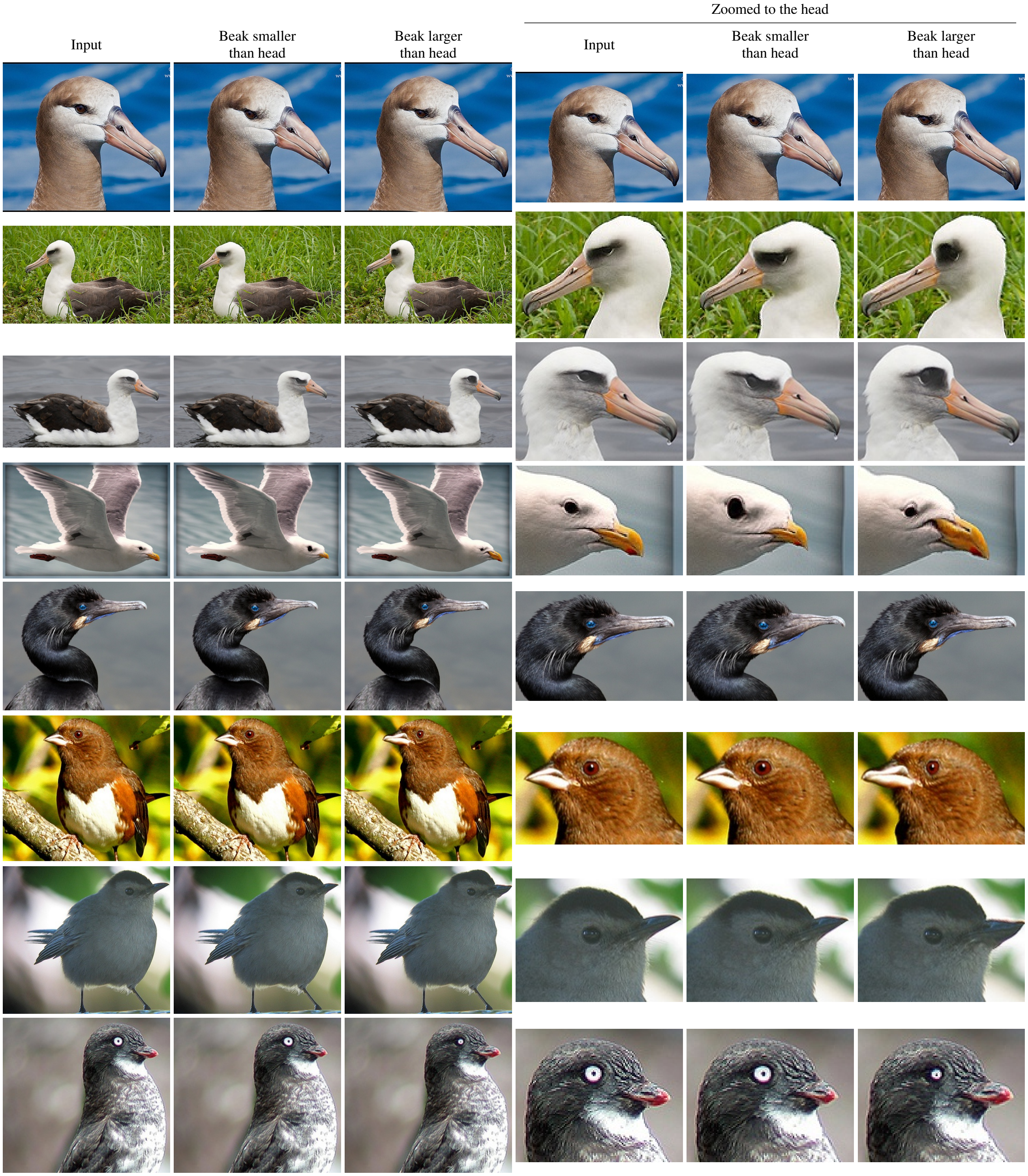

Figure 4: Additional results from our model on test images from the Cub200 dataset. The model attempts to transfer the attribute (relative beak size) in each column to the input image. For easiness of comparison, a crop of the head area is shown in the last three columns.

5. Partial edits on CelebA

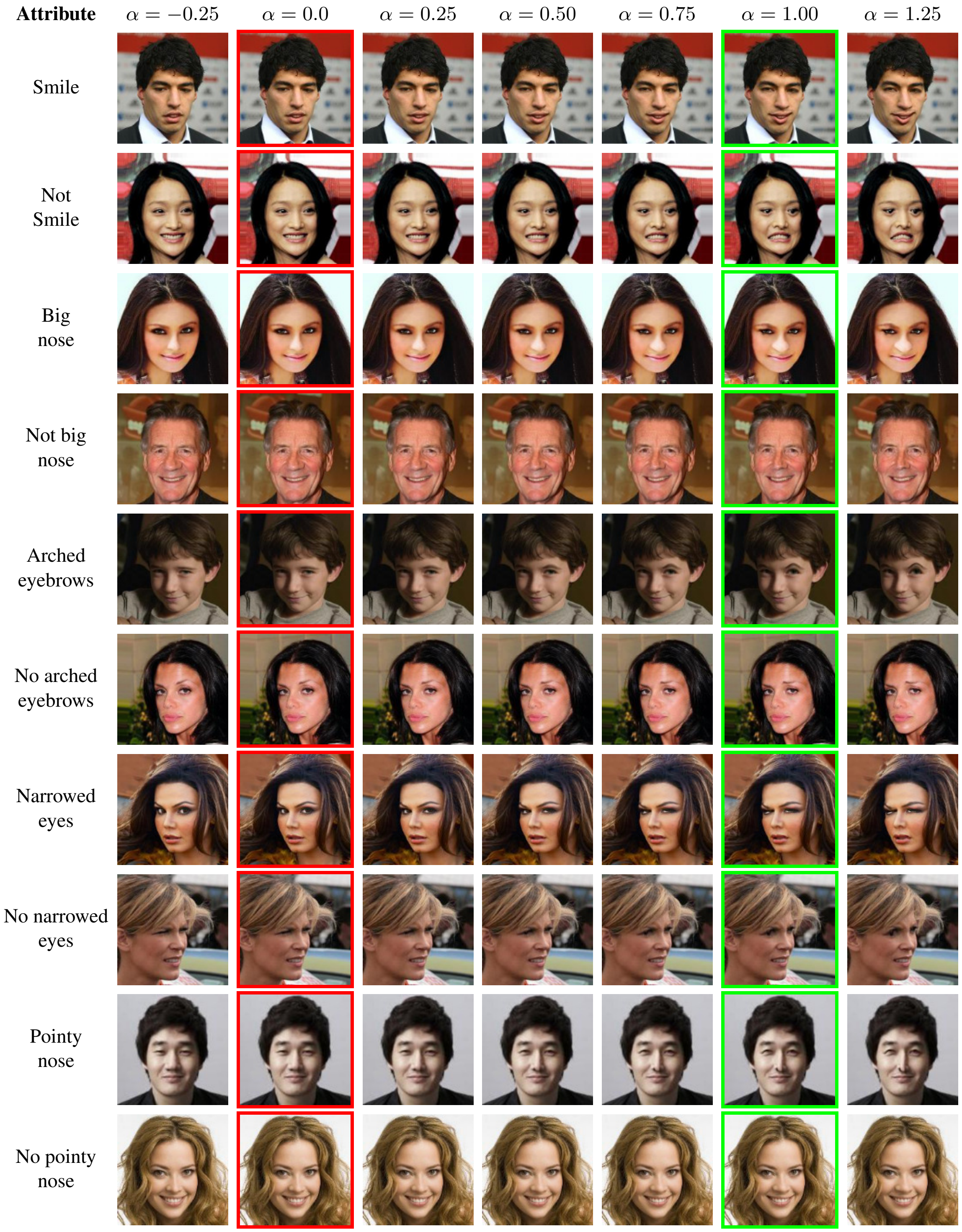

Figure 5: Partial editing with our model, for the the attribute indicated in the first column. A single warp is generated by our model, which is interpolated and extrapolated by scaling the magnitude of its values by α. The input image, $\alpha = 0$, is progressively edited in both directions. A red box denotes the input image, and a green one the output of the generator without α scaling. Please see supplemental videos demonstrating animated edits.

6. Stretch maps on CelebA

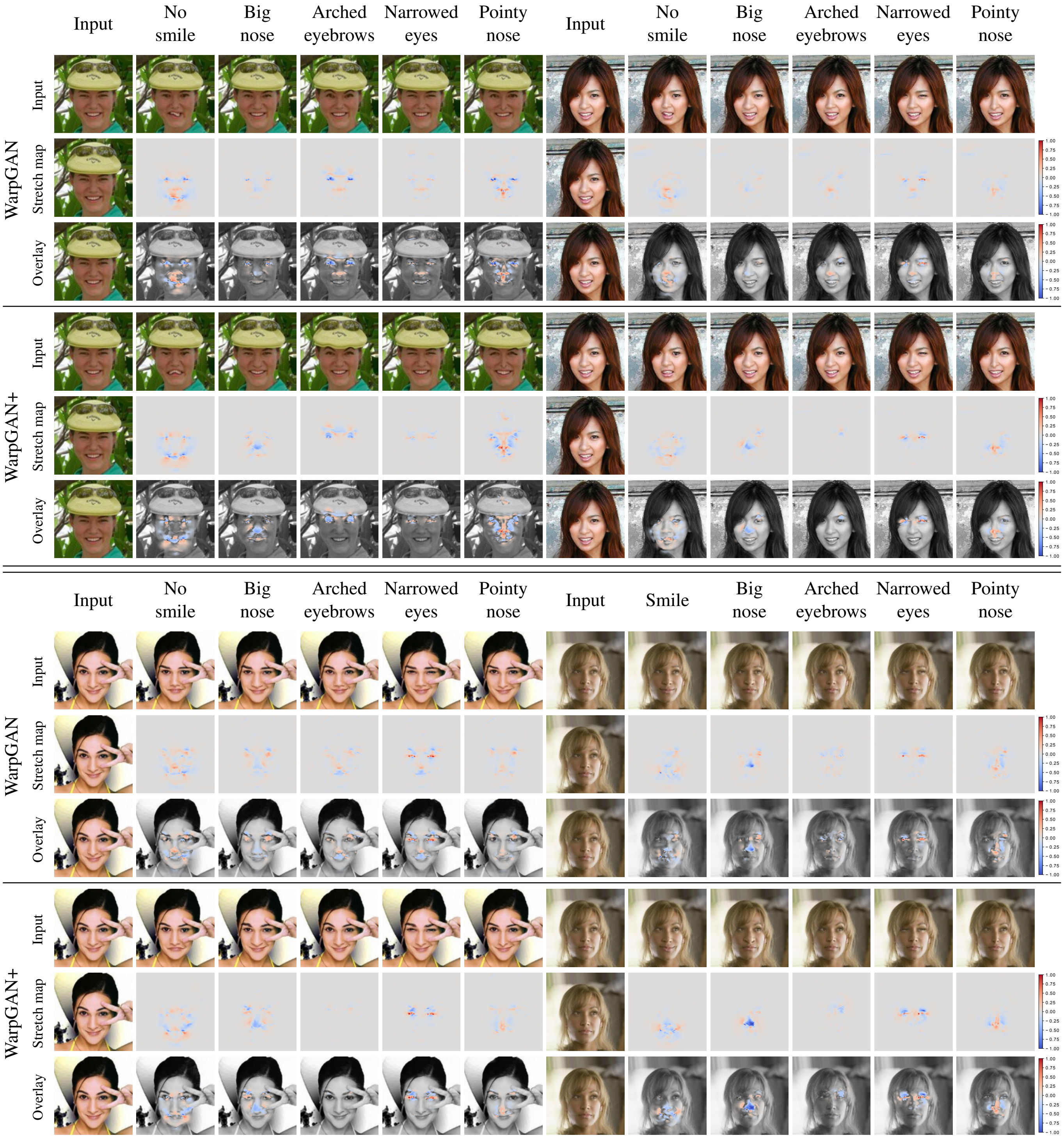

Figure 6: Stretch maps computed from the warp fields, for WarpGAN and WarpGAN+. The log determinant of the Jacobian of the warp is shown, where blue indicates stretching and red corresponds to squashing. Our binary label transformation scheme (WarpGAN+) leads to correctly localized edits.

7. Ablation study

7.1. Effect of each loss

In this section we evaluate the performance of the model after removing each of the losses, where “(w/o) Cycle” corresponds to removing L_{c}, “(w/o) Smooth” corresponds to removing L_{s}, “(w/o) Cls” corresponds to removing L_{cls}, “(pixel) Cycle” corresponds to using eq. 4 instead of eq. 8 in the paper, and “(w/o) Adv” corresponds to removing L_{adv} and L_{gp}.

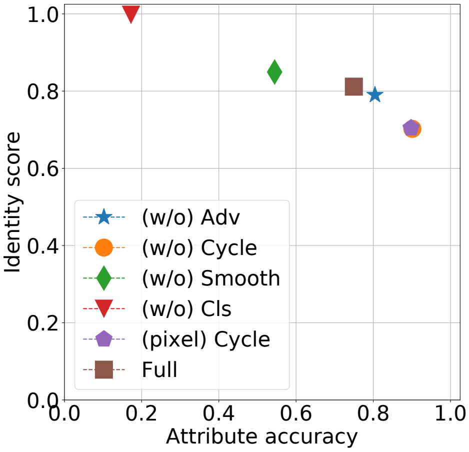

Figure 7: Presence of the edited attribute (x-axis) vs face re-identification score (y-axis), higher is better. Removing each loss in our model has a detrimental effect in either accuracy or identity preservation. The adversarial loss seems to have little effect, however, we qualitatively observed that without it, the edited images were less realistic.

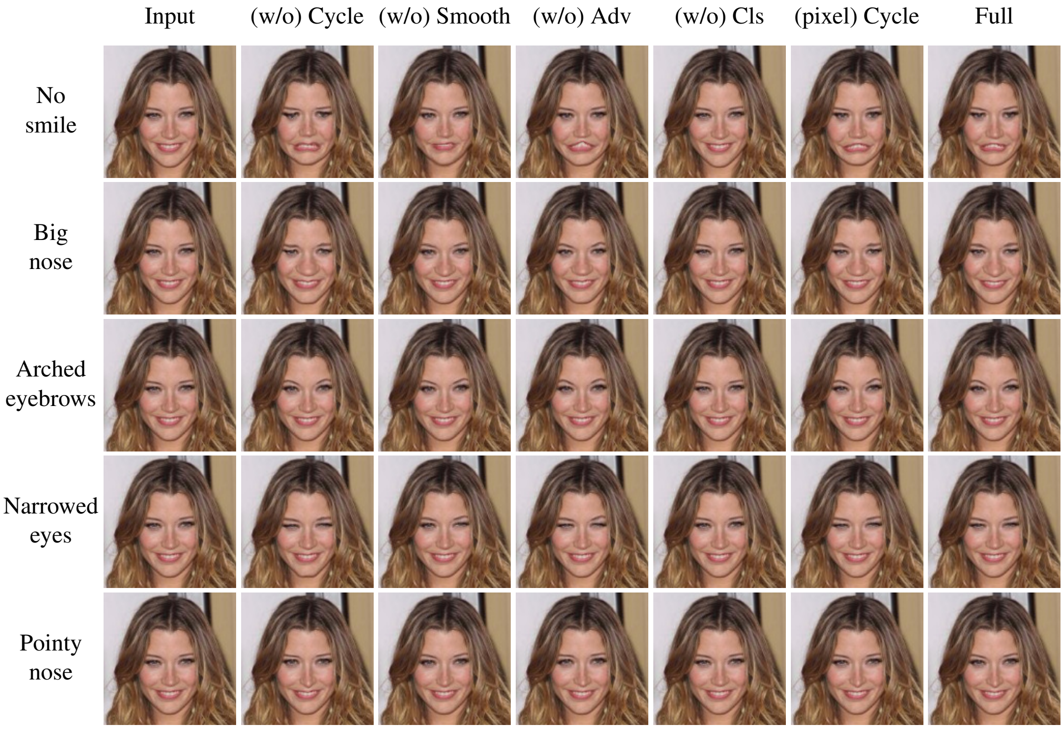

Figure 8: Ablation study, where we remove different losses in our model. For each loss, (w/o) Cycle: significant artifacts are introduced, (w/o) Smooth: leads to poor generalization, (w/o) Adv: unrealistic warps, (pixel) Cycle: exaggerated warps, and (w/o) Cls: trivial solution on the identity transform.

7.2. Effect of α

In this section we quantitatively evaluate the effect of scaling the displacement fields by a scalar α. For this experiment, we take WarpGAN+ trained with $\lambda_{\text{cls}} = 0.25$ and we evaluate the identity score and the attribute accuracy on the test set for different values of α. Results are shown in Fig. 9 for this model, which is denoted as WarpGAN+α. The curve produced by employing different values of α is very similar to the curve in Figure 8 in the paper, which was produced by modifying λ_{cls}. This implies that the model is relatively robust to the choice of λ_{cls}, as a similar effect to changing the value of λ_{cls} used during training can be achieved by choosing an alternative value of α at test time. This is in contrast to previous work, where modifying the strength of the effects requires training a model with the new parameters.

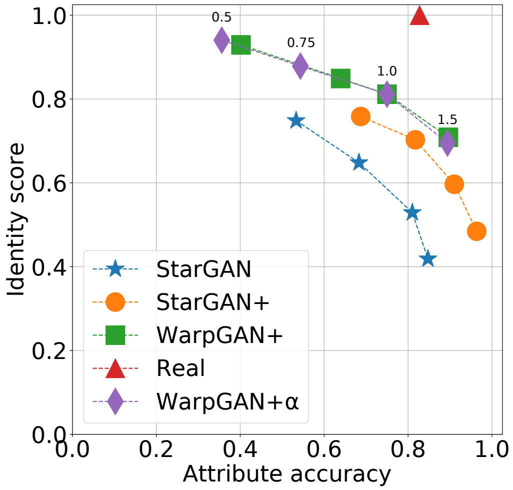

Figure 9: Presence of the edited attribute (x-axis) vs face re-identification score (y-axis), higher is better. For all models except WarpGAN+α, this figure is identical to Fig. 8 in the paper. For WarpGAN+α the value of α is shown on top of each marker. Modifying the α value at test time in our model has a similar effect as training the model with different λ_{cls} values.

8. Face alignment

For the CelebA dataset we use the aligned version provided by the authors, which uses two landmark locations located at the eyes of each subject. Each image is first center-cropped to 178×178, and then resized to 128×128. For the in-the-wild high resolution images from Flickr, an internal face landmark detection network is used to automatically align and resize images to the mean CelebA face at 128×128. The location of the face landmarks used by the network are shown in Fig. 10. For the Cub200 dataset the face alignment to 128 $\times$ 128 uses four landmark locations: the beak, the crown, the forehead and the right eye. If the right eye is not visible, the image is left-right flipped.

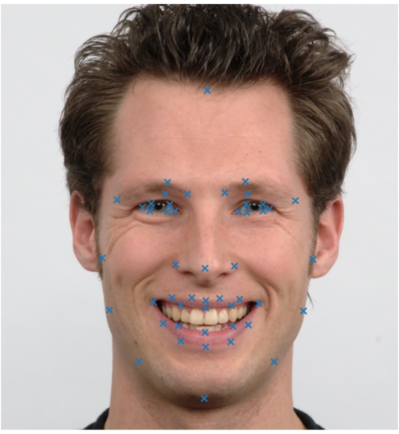

Figure 10: An example of the locations of the 49 face landmarks used for the internal face landmark detection network.

9. Network architectures and training details

The networks were trained on CelebA for 20 epochs and on Cub200 for 1545 epochs (due to the reduced size of this dataset). The Adam optimizer [2] is used with a learning rate of 0.0001, with $\beta_1 = 0.5$ and $\beta_2 = 0.999$.

Our network architectures are based on the StarGAN model. In the generator all transpose convolutions are replaced with bilinear resizing followed by convolution, and instance normalization is replaced by batch normalization. For the discriminator the StarGAN architecture is used without any modifications. In both tables the following notation is used, N is the number of output channels, K is the kernel size, S is the stride size, P is the padding size, and BN is batch normalization. The warping function, T, is implemented with a TensorFlow function during training, and with an OpenCV one for inference:
$T(\mathbf{x}, \mathbf{w}) =$ `tf.contrib.image.dense_image_warp(x, w)`,
$T(\mathbf{x}, \mathbf{w}) =$ `cv2.remap(x, w, interpolation=cv2.INTER_CUBIC)`.

Part	Input → Output Shape	Layer information
Down-sampling	$(h, w, 3+r) \rightarrow (h, w, 64)$	CONV-(N64, K7x7, S1, P3), ReLU, BN
	$(h, w, 64) \rightarrow (\frac{h}{2}, \frac{w}{2}, 128)$	CONV-(N128, K4x4, S2, P1), ReLU, BN
	$(\frac{h}{2}, \frac{w}{2}, 128) \rightarrow (\frac{h}{4}, \frac{w}{4}, 256)$	CONV-(N256, K4x4, S2, P1), ReLU, BN
Bottleneck	$(\frac{h}{4}, \frac{w}{4}, 256) \rightarrow (\frac{h}{4}, \frac{w}{4}, 256)$	Residual Block: CONV-(N256, K3x3, S1, P1), ReLU, BN
	$(\frac{h}{4}, \frac{w}{4}, 256) \rightarrow (\frac{h}{4}, \frac{w}{4}, 256)$	Residual Block: CONV-(N256, K3x3, S1, P1), ReLU, BN
	$(\frac{h}{4}, \frac{w}{4}, 256) \rightarrow (\frac{h}{4}, \frac{w}{4}, 256)$	Residual Block: CONV-(N256, K3x3, S1, P1), ReLU, BN
	$(\frac{h}{4}, \frac{w}{4}, 256) \rightarrow (\frac{h}{4}, \frac{w}{4}, 256)$	Residual Block: CONV-(N256, K3x3, S1, P1), ReLU, BN
	$(\frac{h}{4}, \frac{w}{4}, 256) \rightarrow (\frac{h}{4}, \frac{w}{4}, 256)$	Residual Block: CONV-(N256, K3x3, S1, P1), ReLU, BN
	$(\frac{h}{4}, \frac{w}{4}, 256) \rightarrow (\frac{h}{4}, \frac{w}{4}, 256)$	Residual Block: CONV-(N256, K3x3, S1, P1), ReLU, BN
Up-sampling	$(\frac{h}{4}, \frac{w}{4}, 256) \rightarrow (\frac{h}{2}, \frac{w}{2}, 256)$	Bilinear resize
	$(\frac{h}{2}, \frac{w}{2}, 256) \rightarrow (\frac{h}{2}, \frac{w}{2}, 128)$	CONV-(N128, K4x4, S1, P1), ReLU, BN
	$(\frac{h}{2}, \frac{w}{2}, 128) \rightarrow (h, w, 128)$	Bilinear resize
	$(h, w, 64) \rightarrow (h, w, 64)$	CONV-(N64, K4x4, S1, P1), ReLU, BN
	$(h, w, 64) \rightarrow (h, w, 2)$	CONV-(N2, K7x7, S1, P1)

Table 1: Architecture for the warping network, W, the last layer is the displacement field $\mathbf{w}$, h and w denote the dimensionality of the input image, and r the number of attributes.

Part	Input → Output Shape	Layer information
Down-sampling	$(h, w, 3) \rightarrow (\frac{h}{2}, \frac{w}{2}, 64)$	CONV-(N64, K4x4, S2, P1), Leaky ReLU
	$(\frac{h}{2}, \frac{w}{2}, 64) \rightarrow (\frac{h}{4}, \frac{w}{4}, 128)$	CONV-(N128, K4x4, S2, P1), Leaky ReLU
	$(\frac{h}{4}, \frac{w}{4}, 128) \rightarrow (\frac{h}{8}, \frac{w}{8}, 256)$	CONV-(N256, K4x4, S2, P1), Leaky ReLU
	$(\frac{h}{8}, \frac{w}{8}, 256) \rightarrow (\frac{h}{16}, \frac{w}{16}, 512)$	CONV-(N512, K4x4, S2, P1), Leaky ReLU
	$(\frac{h}{16}, \frac{w}{16}, 512) \rightarrow (\frac{h}{32}, \frac{w}{32}, 1024)$	CONV-(N1024, K4x4, S2, P1), Leaky ReLU
	$(\frac{h}{32}, \frac{w}{32}, 1024) \rightarrow (\frac{h}{64}, \frac{w}{64}, 2048)$	CONV-(N2048, K4x4, S2, P1), Leaky ReLU
Output layer D	$(\frac{h}{64}, \frac{w}{64}, 2048) \rightarrow (\frac{h}{64}, \frac{w}{64}, 1)$	CONV-(N1, K3x3, S1, P1)
Output layer C	$(\frac{h}{64}, \frac{w}{64}, 2048) \rightarrow (1, 1, r)$	CONV-(N(r), K$\frac{h}{64}$x$\frac{w}{64}$, S1, P0)

Table 2: Architecture for the discriminator and the classifier networks, D and C. The kernel weights in the down-sampling layers are shared by D and C.

10. Quantitative results: details

10.1. Accuracy vs identity preservation

In this section we give additional detail about the face re-identification network. We also provide attribute accuracy values and identity scores per attribute for the models used in the paper, namely, for StarGAN and StarGAN+ trained with $\lambda_{\text{cls}} = 0.25$ and for WarpGAN+ with $\lambda_{\text{cls}} = 2.00$.

10.1.1 Re-identification network

For the face re-identificaiton scores, presented in Fig. 8 in the paper, we use a Facenet model pretrained on the MS-Celeb-1M dataset [1]. This dataset consists of 10 million images and 100k unique identities. As both CelebA and MS-Celeb-1M were collected from publicly available Internet images, we expect some overlap between both datasets. This pretrained model is provided by the authors and is publicly available at `https://github.com/davidsandberg/facenet`.

Model	**Smiling**	**Big nose**	**Arched eyebrows**	**Narrowed eyes**	**Pointy nose**	**Mean**
StarGAN	0.65	0.60	0.64	0.66	0.68	0.65
StarGAN+	0.72	0.66	0.67	0.78	0.69	0.70
WarpGAN+	**0.83**	**0.73**	**0.81**	**0.87**	**0.82**	**0.81**
Real	1.00	1.00	1.00	1.00	1.00	1.00

Table 3: Quantitative comparison of the re-identification score on real and generated images on the CelebA dataset evaluated with the face re-identification network, higher is better.

10.1.2 Attribute classification accuracy

Model	**Smiling**	**Big nose**	**Arched eyebrows**	**Narrowed eyes**	**Pointy nose**	**Mean**
StarGAN	0.84	0.60	0.69	0.65	0.62	0.68
StarGAN+	**0.92**	**0.73**	**0.87**	**0.75**	**0.82**	**0.82**
WarpGAN+	0.72	0.72	0.83	0.74	0.74	0.75
Real	0.92	0.81	0.82	0.88	0.72	0.83

Table 4: Quantitative comparison of the attribute classification accuracy on real and generated images on the CelebA dataset evaluated with a separate classification network, higher is better.

10.2. User study

In the user study, for both experiments, to evaluate the reliability of the workers, a number of easy to classify images were mixed with the data, and used as a control. Workers needed to give the right answer to at least 90% of the control images for their data to be considered reliable. Images with fewer than 3 annotations are discarded, as they are considered unreliable data. Finally, a simple majority voting scheme was used to determine the classification of each image.

For the experiment evaluating realism, typical failure cases for all models were shown to the workers before commencing the task, as examples of fake images. For the evaluation of the presence of the target attribute, to guide the workers, curated examples from training data edited with our model were shown to highlight the differences between the attributes.

Some images in the CelebA dataset contain border artifacts due to the alignment process that the authors used for the aligned version of the dataset. In order to get more reliable results, none of these images were included in the pool of 250 images used for the study.

10.2.1 Attribute classification accuracy

Model	Smiling	Big nose	Arched eyebrows	Narrowed eyes	Pointy nose	Mean
StarGAN	**0.85**	0.84	0.75	0.83	0.76	0.81
StarGAN+	**0.85**	0.84	**0.89**	0.86	0.83	**0.86**
WarpGAN+	0.63	**0.92**	0.83	**0.89**	**0.88**	0.84
Real	0.88	0.64	0.74	0.56	0.36	0.63

Table 5: Quantitative comparison of the attribute classification accuracy on real and generated images on the CelebA dataset evaluated with a user study, higher is better.

10.2.2 Realism accuracy

Model	Smiling	Big nose	Arched eyebrows	Narrowed eyes	Pointy nose	Mean
StarGAN	0.40	0.52	0.62	0.41	0.74	0.52
StarGAN+	0.37	0.40	0.59	0.38	0.60	0.46
WarpGAN+	**0.42**	**0.64**	**0.79**	**0.57**	**0.82**	**0.65**
Real	0.97	0.89	0.98	0.96	0.94	0.95

Table 6: Quantitative comparison of image realism both on real and generated images on the CelebA dataset evaluated with a user study, higher is better.

References

[1] Yandong Guo, Lei Zhang, Yuxiao Hu, Xiaodong He, and Jianfeng Gao. MS-Celeb-1M: A dataset and benchmark for large scale face recognition. In *European Conference on Computer Vision*. Springer, 2016. 10

[2] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *ICLR*, 2015. 9