Lifespan Age Transformation Synthesis Supplementary Material

Roy Or-El[1], Soumyadip Sengupta[1], Ohad Fried[2],
Eli Shechtman[3], and Ira Kemelmacher-Shlizerman[1]

[1]University of Washington [2]Stanford University [3]Adobe Research

1 Ethics and Bias Statement

1.1 Intended Use

- This algorithm is designed to hallucinate the aging process and produce an approximation of a person's appearance throughout his/her/their lifespan.
- The main use cases of this method are for art and entertainment purposes (CGI effects, Camera filters, etc.). This method might also be useful for more critical applications, e.g. approximating the appearance of missing people. However, we would like to stress that as a non perfect data-driven method, results might be inaccurate and biased. The output of our method should be critically analyzed by a trained professional, and not be treated as an absolute ground truth.
- **The results of this method should not be used as grounds for detention/arrest of a person or as any other form of legal evidence under any circumstances.**

1.2 Algorithm and Data Bias

We have devoted considerable efforts in our algorithm design to preserve the identity of the person in the input image, and to minimize the influence of the inherent dataset biases on the results. These measures include:

1. Designing the identity encoder architecture to preserve the local structures of the input image.
2. Including training losses that were designed to maintain the person's identity.
   - Latent Identity loss: encourages identity features that are consistent across ages.
   - Cycle loss: drives the network to reproduce the original image from any aged output.
   - Self-reconstruction loss: makes the network learn to reconstruct the input when the target age class is the same as the source age class.
3. The FFHQ dataset contains gender imbalance within age classes. To prevent introducing these biases in the output, e.g. producing male facial features for females or vice versa, we have trained two separate models, one for males and one for females. The decision of which model to apply is left for the

user. We acknowledge that this design choice restricts our algorithm from simulating the aging process of people whose gender is non-binary. Further work is required to make sure future algorithms will be able to simulate aging for the entire gender spectrum.

Despite these measures, the network might still introduce other biases that we did not consider when designing the algorithm. If you spot any bias in the results, please reach out to help future research!

2 Networks Architecture

Our framework consists of a generator, which contains the identity encoder, mapping network and the decoder, an age encoder and a discriminator. We describe the architecture of each component below.

Identity encoder. The identity encoder contains a 7×7 convolution layer that processes the input image. That layer is followed by two 3×3 2-strided convolution layers that downsample the feature maps and four residual blocks [1] that produce the final identity features. Each convolution layer is followed by Pixel-norm [3], which we empirically found to produce less artifacts than Instance-norm and ReLU activation. We applied equalized learning rate [3] for each convolution layer. Table 1 shows the Identity encoder architecture.

Mapping network. The mapping network is an 8 layer MLP network. It takes a $50 \times n$ input age code vector, where n is the number of age classes, and outputs a 256 element age latent code. The input is first normalized with Pixel-norm [3]. Each fully connected layer is followed by a Leaky-ReLU activation and Pixel-norm. We omit the Leaky-ReLU activation for the last layer. We applied equalized learning rate [3] for each fully connected layer. The mapping network architecture can be seen in Table 2.

Decoder. Our decoder contains six styled convolution blocks [4] where we use bilinear upsampling in the last two blocks to return to the original image resolution. To reduce droplet artifacts, we replace each 3×3 convolution + AdaIN [2] combination with a modulated convolution block proposed in StyleGAN2 [5], omitting the noise input. Each modulated convolution layer is followed by a Leaky-ReLU activation and Pixel-norm, which we found to further help in reducing the droplet artifacts. The last layer is a 1×1 convolution that maps the final features of each pixel to RGB values. Equalized learning rate is used in all convolution blocks. Details of the decoder architecture are summarized in Table 3.

Age encoder. The age encoder has a 7×7 convolution that takes the input image. It is followed by four 3×3 2-strided convolution layers that downsample the feature maps, and a 1×1 convolution, that produces a feature map with $50 \times n$ output channels. A global average pooling is then applied to generate the age code vector. Each convolution layer, except for the last one, has a Leaky-ReLU activation. We don't use normalization in the age encoder. Equalized learning rate [3] was applied to each convolution layer. The full age encoder architecture can be found in Table 4.

Layer	Stride	Act.	Norm	Output Shape
Input	–	–	–	$256 \times 256 \times 3$
Conv. 7×7	1	ReLU	Pixel	$256 \times 256 \times 64$
Conv. 3×3	2	ReLU	Pixel	$128 \times 128 \times 128$
Conv. 3×3	2	ReLU	Pixel	$64 \times 64 \times 256$
Res. Block	1	ReLU	Pixel	$64 \times 64 \times 256$
Res. Block	1	ReLU	Pixel	$64 \times 64 \times 256$
Res. Block	1	ReLU	Pixel	$64 \times 64 \times 256$
Res. Block	1	ReLU	Pixel	$64 \times 64 \times 256$

Table 1: Identity encoder architecture.

Layer	Act.	Norm	Output Shape
Age code	–	Pixel	$50 \times n$
Linear	LReLU	Pixel	256
Linear	LReLU	Pixel	256
Linear	LReLU	Pixel	256
Linear	LReLU	Pixel	256
Linear	LReLU	Pixel	256
Linear	LReLU	Pixel	256
Linear	LReLU	Pixel	256
Linear	–	Pixel	256

Table 2: Mapping network architecture.

Layer	Act.	Norm	Output Shape
Idenity Features	–	–	$64 \times 64 \times 256$
Styled Conv.	LReLU	Pixel	$64 \times 64 \times 256$
Styled Conv.	LReLU	Pixel	$64 \times 64 \times 256$
Styled Conv.	LReLU	Pixel	$64 \times 64 \times 256$
Styled Conv.	LReLU	Pixel	$64 \times 64 \times 256$
Styled Conv.	LReLU	Pixel	$64 \times 64 \times 128$
Upsamle	–	–	$128 \times 128 \times 128$
Styled Conv.	LReLU	Pixel	$128 \times 128 \times 64$
Upsamle	–	–	$256 \times 256 \times 64$
Conv. 1×1	Tanh	–	$256 \times 256 \times 3$

Table 3: Decoder architecture.

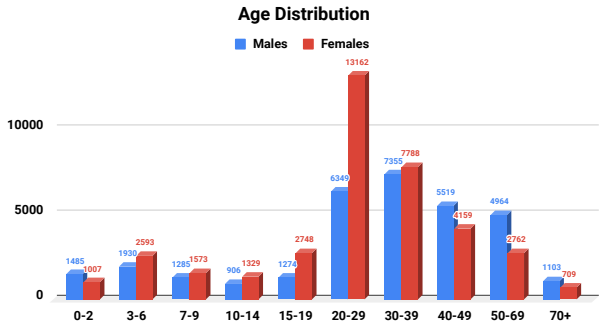
Layer	Stride	Act.	Output Shape
Input	–	–	$256 \times 256 \times 3$
Conv. 7×7	1	LReLU	$256 \times 256 \times 64$
Conv. 3×3	2	LReLU	$128 \times 128 \times 128$
Conv. 3×3	2	LReLU	$64 \times 64 \times 256$
Conv. 3×3	2	LReLU	$32 \times 32 \times 512$
Conv. 3×3	2	LReLU	$16 \times 16 \times 1024$
Conv. 1×1	1	–	$16 \times 16 \times (50 \times n)$
Global Pooling	–	–	$1 \times 1 \times (50 \times n)$

Table 4: Age encoder architecture.

Layer	Act.	Norm	Output Shape
Input	–	–	$256 \times 256 \times 3$
Conv. 1×1	LReLU	–	$256 \times 256 \times 64$
Conv. 3×3	LReLU	–	$256 \times 256 \times 64$
Conv. 3×3	LReLU	–	$256 \times 256 \times 128$
Downsample	–	–	$128 \times 128 \times 128$
Conv. 3×3	LReLU	–	$128 \times 128 \times 128$
Conv. 3×3	LReLU	–	$128 \times 128 \times 256$
Downsample	–	–	$64 \times 64 \times 256$
Conv. 3×3	LReLU	–	$64 \times 64 \times 256$
Conv. 3×3	LReLU	–	$64 \times 64 \times 512$
Downsample	–	–	$32 \times 32 \times 512$
Conv. 3×3	LReLU	–	$32 \times 32 \times 512$
Conv. 3×3	LReLU	–	$32 \times 32 \times 512$
Downsample	–	–	$16 \times 16 \times 512$
Conv. 3×3	LReLU	–	$16 \times 16 \times 512$
Conv. 3×3	LReLU	–	$16 \times 16 \times 512$
Downsample	–	–	$8 \times 8 \times 512$
Conv. 3×3	LReLU	–	$8 \times 8 \times 512$
Conv. 3×3	LReLU	–	$8 \times 8 \times 512$
Downsample	–	–	$4 \times 4 \times 512$
Minibatch Stdev.	–	–	$4 \times 4 \times 513$
Conv. 3×3	LReLU	–	$4 \times 4 \times 512$
Conv. 4×4	LReLU	–	$1 \times 1 \times n$

Table 5: Discriminator architecture.

Discriminator. We use the StyleGAN discriminator [4] architecture with minibatch standard deviation [3]. The first layer is a 1×1 convolution layer that generates a 64 channel feature map for each input pixel. This is followed by twelve 3×3 convolution layers [4], we downsample the feature map after every other 3×3 block (6 times overall). After that we apply minibatch discrimination followed by a 3×3 convolution block and 4×4 convolution block with n output channels in order to discriminate multiple classes as suggested by Liu *et al.* [7]. Leaky ReLU activations and Equalized learning rate are used in all convolution layers. We do not use normalization in the discriminator. Table 5 shows the detailed discriminator architecture.



(a)

Age Class	Males	Females
0–2	1237	804
3–6	1631	2169
7–9	1005	1234
15–19	930	1957
30–39	5512	5848
50–69	3917	2054

(b)

Fig. 1: FFHQ-Aging dataset details. Left: age distributions for males and females for the raw dataset. Right: number of training images for each anchor age class after pruning. The majority of training classes contain more than 1,000 images, which we found sufficient for training our model.

3 FFHQ-Aging Dataset Details

Fig. 1a shows the age distribution of images in the raw FFHQ-Aging dataset for males and females. Figure 1b shows the number of training images for each age class after the data cleaning process described in Sec. 4 of the main paper.

To align the images, we use the same data alignment technique as Karras *et al.* [3] (see Fig. 8e in their paper), which was also used to align the original FFHQ dataset. We mirror pad the image boundaries and then blur them. Then, we use the eyes and mouth landmark locations to select an oriented crop area according to

$$
\begin{aligned}
x' &= e_r - e_l \\
y' &= \frac{1}{2}(m_r + m_l) - \frac{1}{2}(e_r + e_l) \\
c &= \frac{1}{2}(e_r + e_l) - 0.1 \cdot y' \\
s &= \max(4.0 \cdot |\text{Normalize}(x')|, 4.4 \cdot |\text{Normalize}(y')|) \\
x &= \frac{s}{2} \cdot (\text{Normalize}(x' - \text{Rotate90}(y'))) \\
y &= \text{Rotate90}(x) \\
\text{Box} &= [c - x - y, c - x + y, c + x + y, c + x - y]
\end{aligned}
$$

Where e_l, e_r are the landmarks for the left and right eyes respectively, m_l, m_r are the landmarks for left and right corners of the mouth, "Normalize" is vector normalization, s is the size of the box, and c is the center of the cropping box. In order to make sure we obtain the full head that also includes the neck, we took slightly larger crops then the original FFHQ dataset, our scale factor for y' is 4.4 as opposed to 3.6 which was used originally.

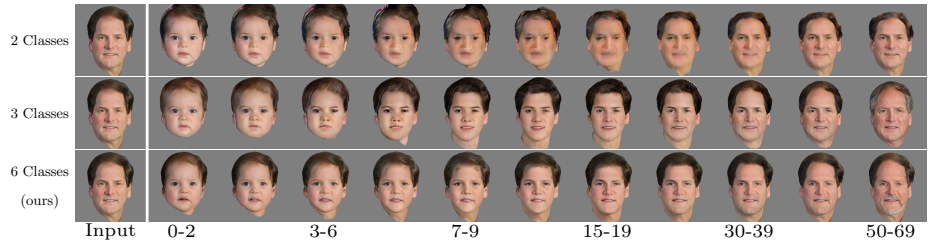


Fig. 2: Anchor classes ablation study. We show latent interpolation on models trained on 2 anchor classes (top row), 3 anchor classes (middle row) and 6 anchor classes (bottom row). Increasing the number of anchor classes greatly improves the framework's ability to generate high quality age transformations over full lifespan.

4 Additional Results

4.1 Continuous Age Transformations

We generated continuous lifespan age transformations by interpolating 24 output images between each neighboring age class anchors. See attached videos and Figures 7 and 8 for the results.

4.2 Ablation Studies

We performed two ablation studies in order to prove our main claims. In the first study we show the importance of using multiple age classes as anchors in order to learn a latent space $\mathcal{W}_{age}$ that will allow for continuous age transformations. We trained two additional models, one with age classes 0–2 & 50–69 as the only anchors and one with age classes 0–2, 15–19 & 50–69 as the anchors. We then generated full lifespan transformation of 11 images from each model by interpolating missing anchor classes when needed along with interpolating one output image between each two base classes. Fig. 2 shows how additional anchor classes are crucial in creating reliable and plausible lifespan age transformations.

In the second study, we examined importance of our design choices in constructing the input age vector code space $\mathcal{Z}$. We show the connection between the structure of $\mathcal{Z}$ to the ability of the age latent space $\mathcal{W}_{age}$ to span all possible ages. Specifically, we show the importance of using multiple vector elements to represent each age class as well as the importance of adding noise to the one-hot input signal. We trained two additional models on all 6 anchor classes, one with 50 elements per age class, but with no added noise, and one with a single element per age class and no added noise. In Fig. 3 we can see that although the anchor classes are always well represented within the latent space, both number of elements per age class and added noise, are important parts to ensure the continuity of $\mathcal{W}_{age}$ and high image quality.

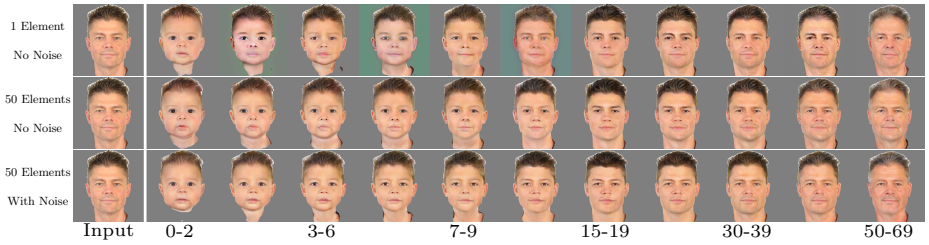


Fig. 3: Age class representation ablation study. We show latent interpolation on models trained with one-hot representation with 1 element per age class (top row), one-hot representation with 50 elements per age class (middle row) and one-hot representation with 50 elements per age class and added gaussian noise (bottom row). Expanding the number of elements representing each age class allows representation of ages outside the anchor classes. Adding noise, further improves the image quality for interpolated outputs (Zoom in for details).

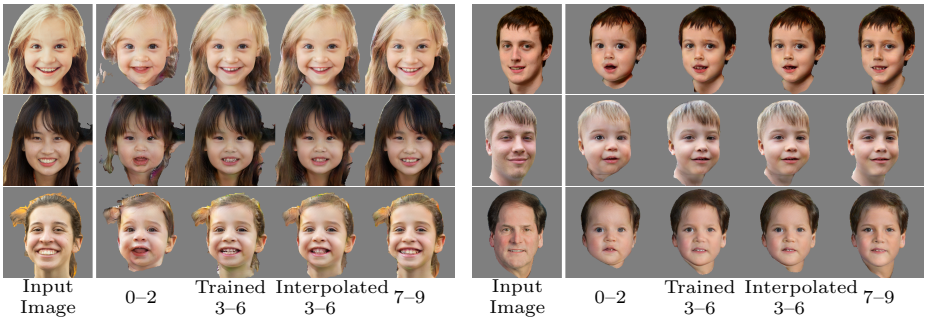


Fig. 4: Linearity of age latent space. We compare the results of the network outputs for the 3–6 class vs. the network outputs for 3–6 class interpolated as the mid point between the 0–2 and 7–9 age latent vectors $0.5 \cdot w_{age}^{0-2} + 0.5 \cdot w_{age}^{7-9}$. The resemblance of the interpolated results to the trained results suggests that age is spanned quasi-linearly in the $\mathcal{W}_{age}$ latent space.

4.3 Generalization Ability

To test our framework ability to generalize, we carried out two experiments. In the first experiment, we tested the generalization ability of the age latent space $\mathcal{W}_{age}$. We produced outputs for the 3–6 age class by interpolating it as the mid point of 0–2 and 7–9 age classes. We fed the decoder a latent age vector $\tilde{w}_{age}^{3-6} = 0.5 \cdot w_{age}^{0-2} + 0.5 \cdot w_{age}^{7-9}$ and compare the results with the outputs for the trained 3–6 class. As can be seen in Fig. 4, the similarity between the trained results and the interpolated results suggests that the learned age latent space, $\mathcal{W}_{age}$, is approximately linear w.r.t the target age input which contributes to the ability of the framework to generate results outside of the trained age classes.

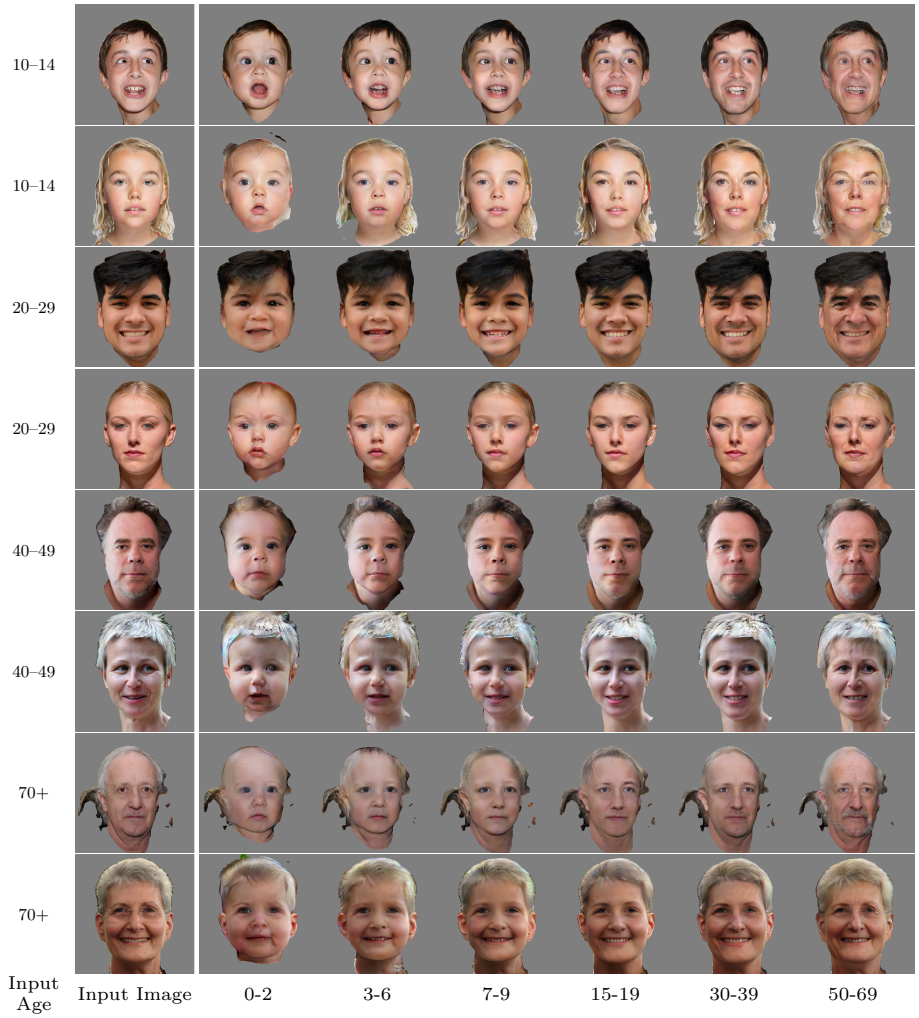


Fig. 5: Results on inputs from untrained age classes. Note that masking artifacts are a result of the segmentation process, and were not caused due to our method.

In the second experiment, we tested the generalization ability of the identity feature space. We feed the network images from the remaining 4 untrained classes in FFHQ-Aging, 10–14, 20–29, 40–49 & 70+. Fig. 5 demonstrates our method's ability to produce high-quality results for unseen face structures from unseen age classes.

4.4 User Studies

The user interface of our user studies is presented in Figure 6. The same UI was used both for the studies in the main paper and in this supplemental doc-

Fig. 6: User study interface. We asked 3 different questions to assess age, identity and overall quality.

Age range:	0–2		3–6		7–9		15–19		30–39		50–69		All	
	[8]	Ours	[8]	Ours	[8]	Ours	[8]	Ours	[8]	Ours	[8]	Ours	[8]	Ours
Same identity ↑	14	**20**	19	**23**	24	24	20	**25**	**24**	22	19	**23**	120	**137**
Age difference ↓	**1.0**	3.4	**2.1**	3.2	**4.5**	5.1	**6.4**	10.3	8.2	**7.4**	13.3	**6.5**	**5.9**	6.0
Overall better ↑	2	**23**	1	**24**	1	**23**	2	**23**	1	**24**	0	**25**	7	**142**

Table 6: User study results vs. IPCGAN [8] that was retrained on our dataset. Our results are better at preserving subject identity, and the two methods are extremely close at age accuracy. Most importantly, when asked which result is better overall, users preferred our results in 95% of the cases (142 out of 150, compared to 7 for IPCGAN and 1 indecisive).

Age range:	0–2		3–6		7–9		15–19		30–39		50–69		All	
	[6]	Ours	[6]	Ours	[6]	Ours	[6]	Ours	[6]	Ours	[6]	Ours	[6]	Ours
Same identity ↑	16	**22**	24	**25**	25	25	**25**	24	**25**	24	**25**	24	140	**144**
Age difference ↓	**4.0**	4.4	15.7	**6.2**	19.8	**9.5**	17.5	**12.3**	13.3	**7.0**	23.1	**7.7**	15.6	**7.8**
Overall better ↑	5	**20**	6	**18**	3	**20**	3	**20**	3	**21**	1	**24**	21	**123**

Table 7: User study results vs. STGAN [6] that was retrained on our dataset. Our results are better at preserving subject identity, and have better age accuracy. Most importantly, when asked which result is better overall, users preferred our results in 82% of the cases (123 out of 150, compared to 21 for STGAN and 6 indecisive).

ument. In addition to the main paper user studies, we also wanted to verify that our results are not solely due to a better dataset. To this end, we retrained IPCGAN [8] and STGAN [6] on our data.

In the following studies, we evaluate the results of 25 randomly selected photos on each of the 6 age classes, repeating each question 3 times, for a total of 2250 individual answers per user study. Note that in these studies we can compare all 6 age groups, whereas in our other user studies we were limited by the choice to use the authors' pre-trained models which were not available for all ages.

User study results are in Tables 6 and 7. Indeed, we see that even when retrained on our data, there is a significant performance gap between our results and previous works [6,8]. Our results are better at identity preservation, and either better or on-par in age accuracy. As explained in the main text, since

overall quality is determined by both these factors and others such as image quality, we asked users which result is better overall. Our results were selected as better in 82% (vs. StGAN) and 95% (vs. IPCGAN) of the cases.

References

1. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016) 2
2. Huang, X., Belongie, S.: Arbitrary style transfer in real-time with adaptive instance normalization. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1501–1510 (2017) 2
3. 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) 2, 3, 4
4. Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4401–4410 (2019) 2, 3
5. Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., Aila, T.: Analyzing and improving the image quality of StyleGAN. CoRR **abs/1912.04958** (2019) 2
6. Liu, M., Ding, Y., Xia, M., Liu, X., Ding, E., Zuo, W., Wen, S.: Stgan: A unified selective transfer network for arbitrary image attribute editing. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3673–3682 (2019) 8
7. Liu, M.Y., Huang, X., Mallya, A., Karras, T., Aila, T., Lehtinen, J., Kautz, J.: Few-shot unsupervised image-to-image translation. arXiv preprint arXiv:1905.01723 (2019) 3
8. Wang, Z., Tang, X., Luo, W., Gao, S.: Face aging with identity-preserved conditional generative adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 7939–7947 (2018) 8

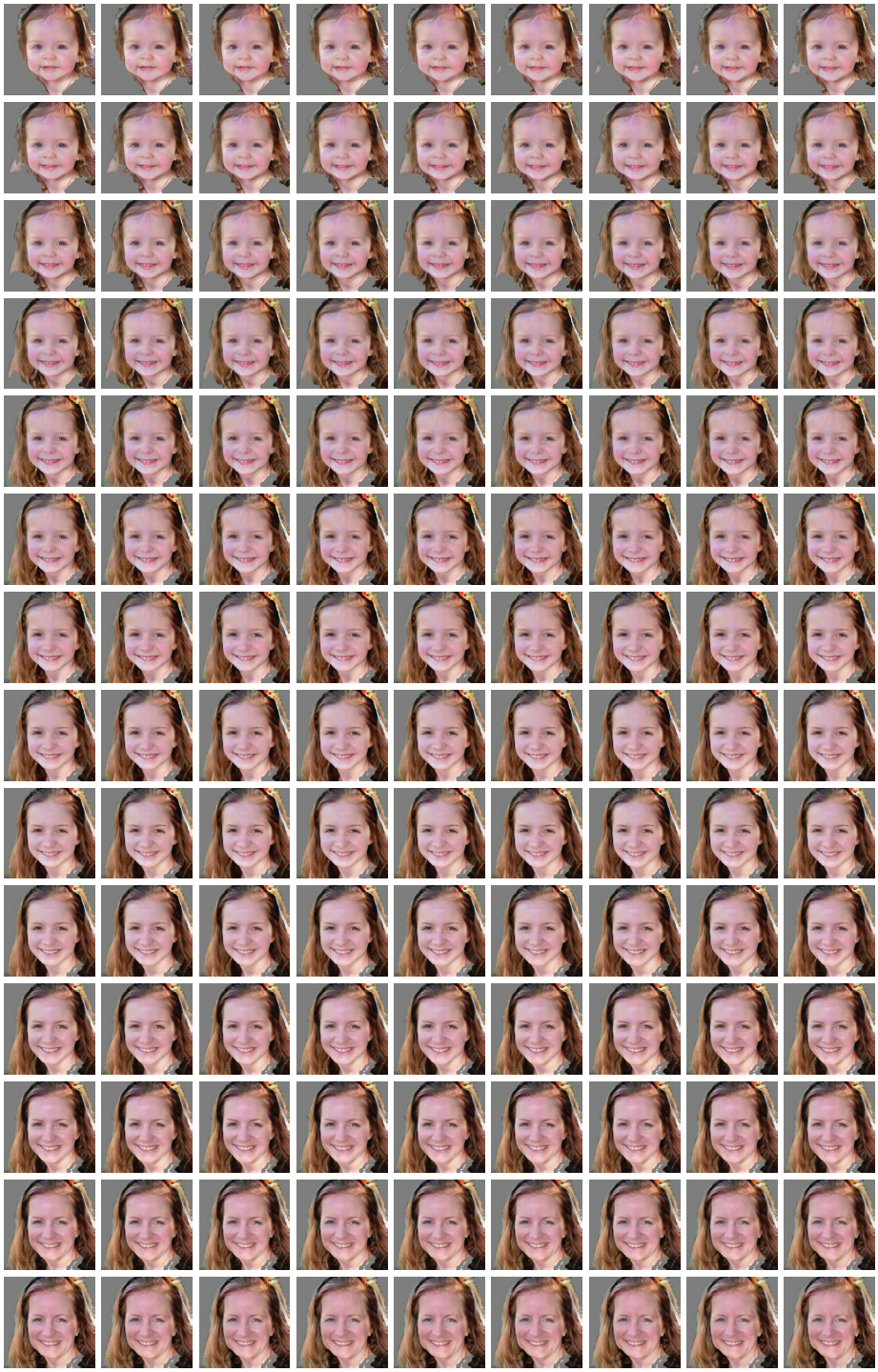


Fig. 7: Full lifespan transformation. Also see supplemental videos.

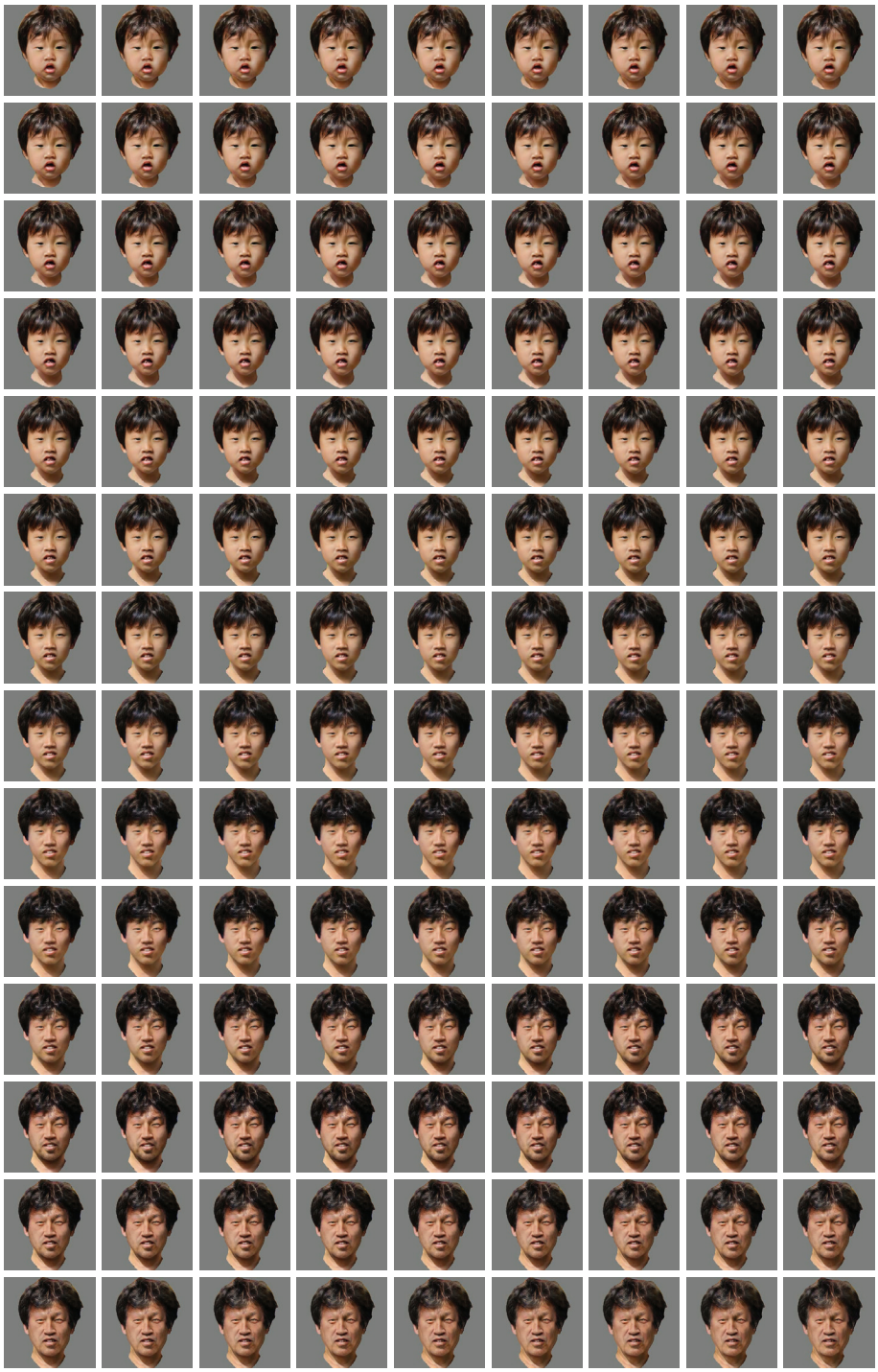


Fig. 8: Full lifespan transformation. Also see supplemental videos.