Intuitive, Interactive Beard and Hair Synthesis with Generative Models – Supplementary Material

1. Interactive Editing

Fig. 1 shows the interface of our system. As described in the paper (Sec. 6), we provide tools for synthesizing a coarse initialization of the target hair style given only the user-drawn mask and selected color; separately manipulating the color and vector fields used to automatically extract guide strokes of the appropriate shape and color from this initial estimate; and drawing, removing, and changing individual strokes to make local edits to the final synthesized image. See Fig. 2 for an example of iterative refinement of an image using our provided input tools for mask creation and individual stroke drawing with the corresponding output, and Fig. 3 for an example of vector/color field editing, that easily changes the structure and color of the guide strokes. Also see Fig. 4 in the paper and the example sessions in the supplementary video for examples of conditional inpainting and other editing operations.

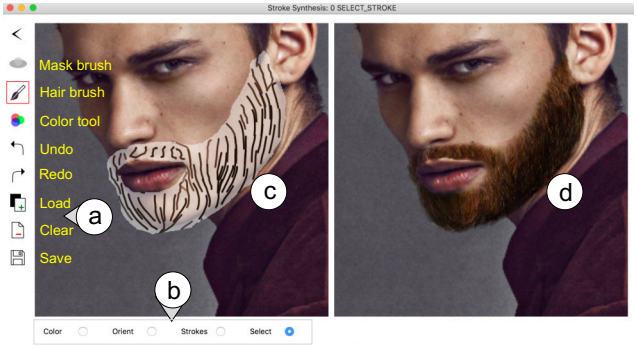


Figure 1: The user interface, consisting of a global (a) and contextual (b) toolbars, and a main (c) and result preview (d) canvases.

Fig. 4 shows an example of how the overall structure of a synthesized hair style can be changed by making adjustments to the structure of the user-provided guide strokes. By using strokes with the overall colors of those in row 1, column 1, but with different shapes, such as the smoother and more coherent strokes as in row 2, column 1, we can generate a correspondingly smooth and coherent hair style, (row 2, column 2).

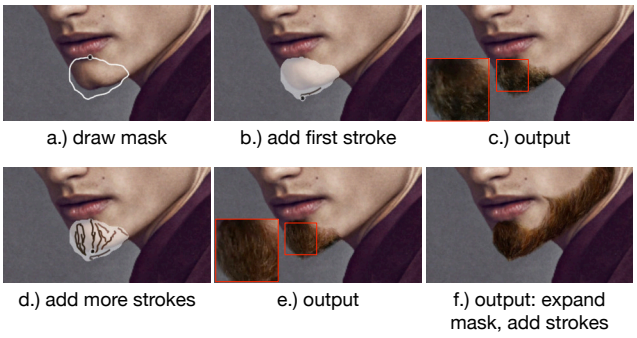


Figure 2: An example of interactive facial hair creation from scratch on a clean-shaven target image. After creating the initial mask (a), the user draws strokes in this region that define the local color and shape of the hair (b-e). While a single stroke (b) has little effect on much of the masked region (c), adding more strokes (d) results in more control over the output (e). The user can also change the shape of the mask and add additional strokes (f) to adjust the overall hair style.

Figure 3: Changes to the vector and color fields cause corresponding changes in the guide strokes, and also the final synthesized results.

Facial hair reference database We provide a library of sample images from our facial hair database that can be used as visual references for target hair styles. The user can select colors from regions of these images for the initial color mask, individual strokes and brush-based color field editing. This allows users to easily choose colors that represent the overall and local appearance of the desired hair style. Users may also copy and paste selected strokes from these images directly into the target region, so as to directly

emulate the appearance of the reference image. This can be done using either the color of these selected strokes in the reference image, or merely their shape with the selected color and transparency settings so as to emulate the local structure of the reference image within the global structure and appearance of the target image.

Figure 4: Subtle changes to the structure and appearance of the strokes used for final image synthesis cause corresponding changes in the final synthesized result.

2. Scalp and Facial Hair Synthesis Results

As described in Sec. 7 and displayed in Fig. 8 of the main text, we found that introducing segmented scalp hair with corresponding guide strokes (automatically extracted as described in Sec. 5 of the main text) allows for synthesizing high-quality scalp hair and plausible facial hair with a single pair of trained networks to perform initial synthesis, followed by refinement and compositing.

We use 5320 images with segmented scalp hair regions for these experiments. We do this by adding a second end-to-end training stage as described in Sec. 5 in which the two-stage network pipeline is refined using only these scalp hair images. Interestingly, simply using the the real scalp and facial hair dataset simultaneously did not produce acceptable results. This suggests that the multi-stage refinement process we used to adapt our synthetic facial hair dataset to real facial hair images is also useful for further adapting the trained model to more general hair styles.

Fig. 5 portrays several additional qualitative results from these experiments (all other results seen in the paper and supplementary material, with the exception of Fig. 8 in the paper, were generated using a model trained using only the synthetic and real facial hair datasets). As can be seen, we can synthesize a large variety of hair styles with varying structure and appearance for both female (row 1) and male (rows 2-4) subjects using this model, and can synthesize both scalp and facial hair simultaneously (rows 2-4, columns 6-7). Though this increased flexibility in the types of hair that can be synthesized using this model comes with a small decrease in quality in some types of facial hair quite different from that seen in the scalp hair database (*e.g.*, the short, sparse facial hair seen in Fig. 5, row 2, column 7 of the main paper, which was synthesized using a model trained using only facial hair images), relatively dense facial hair such as those portrayed here can still be plausibly synthesized while synthesizing a wide variety of scalp hair styles.

3. Ablation Study Results

We show selected qualitative results from the ablation study described in Table 1 of the main text in Fig. 6. With the addition of the refinement network, our results contain more subtle details and have fewer artifacts at skin boundaries than when using only one network. Adversarial loss adds more fine-scale details, while VGG perceptual loss reduces amount of noisy artifacts. Compared with networks trained with only real image data, our final result has clearer definition for individual hair strands and has a higher dynamic range, which is preferable if users are to perform elaborate image editing. With all these components, our networks produce results of much higher quality than the baseline network of [2].

4. User study

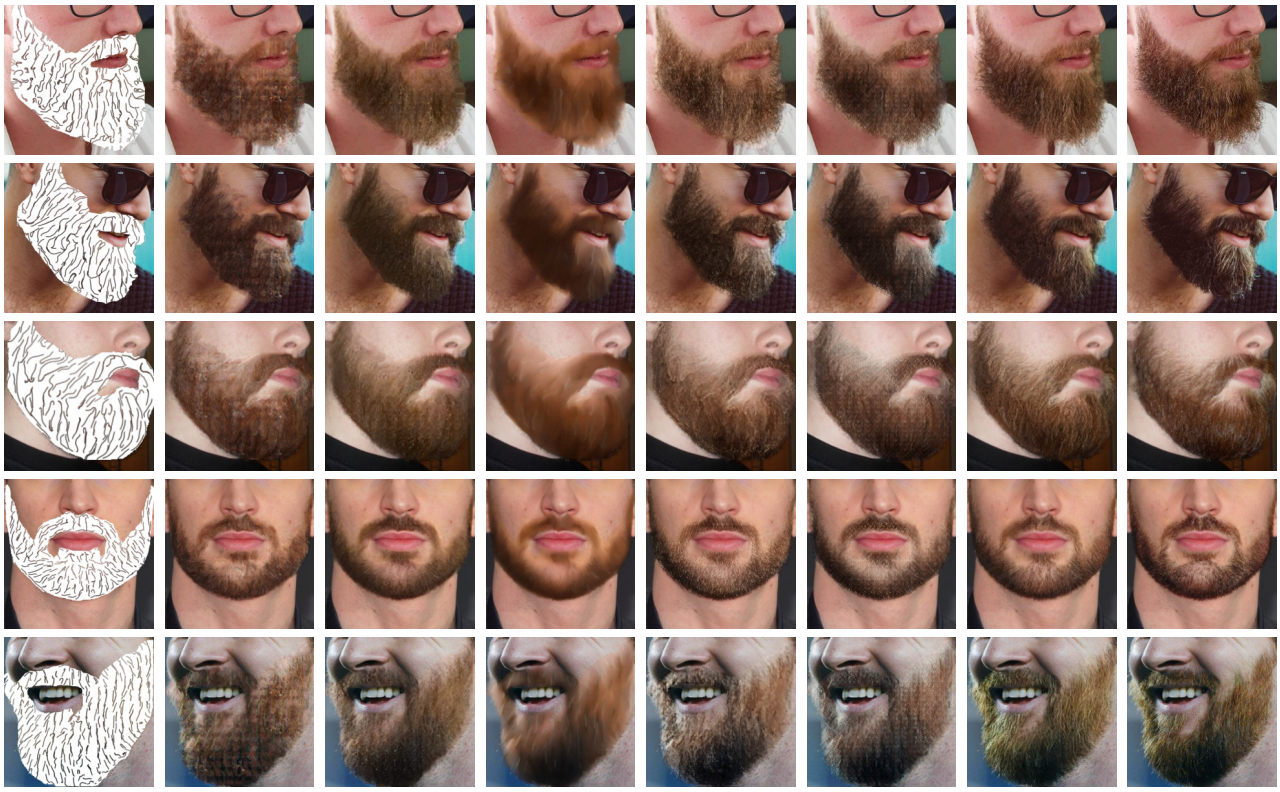
We conducted a preliminary user study to evaluate the usability of our system. The study included 8 users. 1 user was a professional technical artist, while the others were non-professionals, including novices with minimal to moderate prior experience with technical drawing or image editing. When asked to rate their prior experience as a technical artist on a scale of $1-5$, with 1 indicating no prior experience and 5 indicating a professionally trained technical artist, the average score was 3.19.

Procedure The study consisted of three sessions: a warm-up session (approximately 10 min), a target session (15-25 min), and an open session (10 min). The users were then asked to provide feedback by answering a set of questions

Figure 5: Example results for synthesizing scalp hair, both with and without facial hair. Rows 1-2 shows examples for female subjects, while rows 3-5 depict male subjects. For the male subjects, columns 6-7 depict input and output to synthesize facial hair with scalp hair. These results are generated using the same models trained on a combination of facial and scalp hair.

to quantitatively and qualitatively evaluate their experience. In the warm-up session, users were introduced to the input operations and workflow and then asked to familiarize themselves with these tools by synthesizing facial hair on a clean-shaven source image similar to that seen in a reference image. For the target session, the participants were given a new reference portrait image and asked to create similar hair on a different clean-shaven subject via (1) our interface, and (2) Brushables [4]. For Brushables, the user was asked to draw an vector field corresponding to the overall shape and orientation of the facial hair style in the target image. A patch of facial hair taken directly from the target image was used with this input to automatically synthesize the output facial hair. For the open session, we let the participants explore the full functionality of our system and uncover potential usability issues by creating facial hair with arbitrary structures and colors.

Outcome Fig. 7 provides qualitative results from the target session. The users took between $6-19$ minutes to create the target image using our tool. The average session time was 14 minutes. The users required an average of 116 strokes to synthesize the target facial hair style on the source subject. 39 of these strokes were required to draw and edit the initial mask defining the region in which synthesis is performed. The remaining 77 were brush strokes used to edit the color and vector fields used to automatically generate strokes in this region, and to draw the individual strokes used to perform the final refinement. On average a user performed 19 brush strokes to edit the vector field, 17 brush strokes to edit the color field, and drew 41 individual strokes. We note that these numbers include individual strokes deleted by the user if their impact on the resulting image was deemed unsatisfactory. Overall, these numbers indicate, as does the provided feedback, that the color and



(a) Input (b) Isola *et al*. (c) Single Network (d) w/o GAN (e) w/o VGG (f) w/o synth. data (g) Ours, final (h) Ground truth

Figure 6: Qualitative comparisons for ablation study.

brush editing tools were useful in reducing the number of individual strokes required to synthesize the final image. We use the original Brushables implementation for comparison, which does not provide statistics on the number of operations performed by the user, and as such we can not report these statistics for the Brushables session.

Feedback We asked the users to rate our method in terms of ease-of-use and their perceived quality of the final image they created during the target session. On a scale of $1-5$ (higher is better), the users rated our system 3.6 in terms of ease of use, and 4.06 in terms of their synthesized result matching the target facial hair style. When asked to measure how satisfied they were with the result given the amount of time they spent creating it and becoming familiar with the system, the average score was 4.0. Furthermore, 100% of the users preferred our system over Brushables for the task of facial hair editing.

After being introduced to its interface, users spent between $3-5$ minutes (4 minutes on average) working with Brushables to attempt to synthesize the target hair style. While less time was required to synthesize the results with Brushables, the users generally found the results achieved by copying regions of the source texture sample directly into the specified target region to be very unsatisfactory. By simply attempting to create an vector field roughly matching the structure of the target hair style and then synthesizing the result, users had little control over the subtle local details necessary to synthesize a plausible result. Furthermore, as Brushables required approximately 30 seconds to synthesize the entire facial hair style given the complete user-defined vector field, iterative experimentation was far more difficult than when using our approach that allows for immediately visualizing the results of minor editing operations. Thus, users chose not to experiment with the Brushables approach long enough to produce more satisfactory results.

Overall, the participants found our system novel and useful. When asked what features they found most useful, some users commented that they liked the ability to create a rough approximation of the target hairstyle given only a mask and average color. Others strongly appreciated the color and orientation brushes, as these allowed them to separately change the color and structure of the initial estimate, and to change large regions of the image without drawing each individual stroke with the appropriate shape and

Figure 7: Example results generated by participants in our user study, when asked to synthesize a style resembling the target image (top left). No user had prior experience using our interface. Subjects took an average of 14 minutes (and no more than 19 minutes) to create the portrayed images.

color. In contrast, drawing each individual stroke manually was not perceived as especially useful, as it required significantly more effort to experiment with creating and removing individual strokes to produce a combination with the appropriate shape and color to achieve the desired result. However, overall participants were able to achieve reasonable results such as those in Fig. 7 primarily relying on the color and vector field brushes to edit the initial synthesis results produced when selecting the mask shape and color. Relatively few individual strokes were ultimately required to refine the results.

This feedback suggests that the increasing level of granularity enabled by our system (creating a rough initial estimate, modifying the local shape and color, and then refining small details with a few individual strokes) is an effective approach. Furthermore, the participants reported the real-time synthesis and visualization of the generated image allowed for intuitive iterative refinement, which provided them with helpful visual guidance in producing an accurate final result.

During the open session at the end, users enjoyed experimenting with our tools to creatively generate unconventional hairstyles with unusual shapes and colors. However, as many of these styles were well outside the range of natural shapes and colors seen in the images used to train our system, the results were less realistic than those constrained to resemble a more conventional hairstyle.

5. Implementation Details

We now describe how we train the proposed two-stage approach. We train the first stage network first with synthetic then with real data. Then, we train both stages in an end-to-end manner with real images while keeping the losses for both stages.

When training the first network individually, we use an initial learning rate of 0.0002 and momentum of 0.5. The learning rate is halved twice during this training process such that in the final epochs the learning rate is reduced to 0.00005. During end-to-end training of both networks, the initial learning rate is reduced to 0.0001 and a momentum of 0.75 is used. As before, the learning rate is halved twice during training, resulting in a final learning rate of 0.000025.

Both network architectures are fully convolutional and thus can take input images of any resolution. Nevertheless, we scale all our training data to a resolution of 512×512. We train both networks via the Adam optimizer [3] on an NVIDIA Titan X GPU using the Torch framework [1]. We first train each stage of the networks for 50 epochs which takes about 24 hours. Then refine the first network using real image data to train for 25 epochs which takes about 12 hours.

The user interface is designed to allow for input using

either a traditional mouse for novice users, or the tablet and stylus tools used by digital artists. For run-time interaction, passing one image through our network takes a total of 600 milliseconds. We transmit the input image to a server running our networks. The total time between making an update to the input and seeing the corresponding on average thus takes roughly 1.5 seconds. The vector field used for the initial stroke extraction is also performed using CUDA for GPU acceleration, and takes roughly 120 milliseconds for a 512×512 image.

References

[1] Ronan Collobert, Samy Bengio, and Johnny Marithoz. Torch: A modular machine learning software library, 2002. 5

[2] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. *arXiv preprint arXiv:1611.07004*, 2016. 2

[3] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014. 5

[4] Michal Lukáč, Jakub Fiser, Paul Asente, Jingwan Lu, Eli Shechtman, and Daniel Sýkora. Brushables: Example-based edge-aware directional texture painting. *Comput. Graph. Forum*, 34(7):257–267, 2015. 3