Supplementary Material
Paint2Pix: Interactive Painting based Progressive Image Synthesis and Editing

Jaskirat Singh[1,2], Liang Zheng[1], Cameron Smith[2], and Jose Echevarria[2]

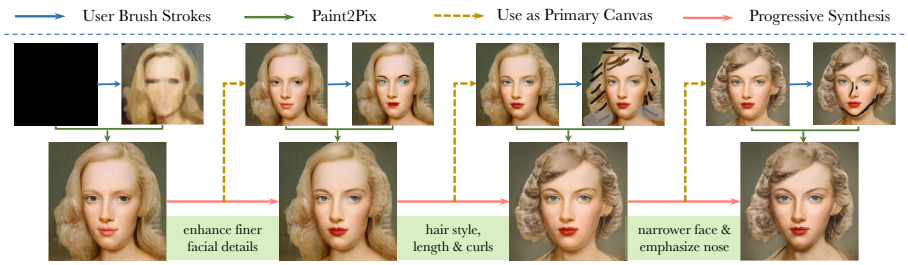
[1] Australian National University
[2] Adobe Research

A Paint2pix for Artistic Content Generation

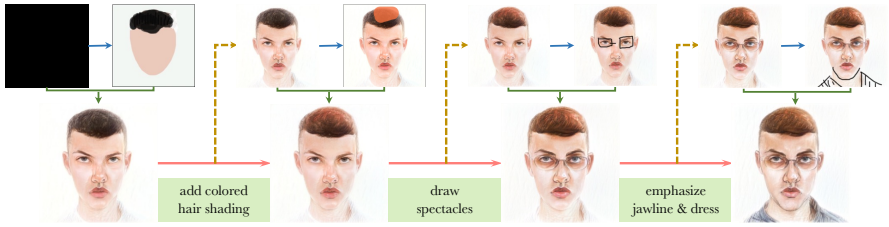
While the main goal of *paint2pix* is to perform photorealistic image synthesis from primitive human paintings and brushstrokes, it can also be used for artistic content generation. In this section, we explore how *paint2pix* along with encoder-bootstrapping [1] would allow a novice artist to generate different forms of professional artistic content (*e.g.*, colored sketches, oil painting, disney cartoons, watercolor paintings, *etc.*) using just few rudimentary brushstroke inputs.

Approach. In particular, we first use the non-adversarial domain adaptation strategy from Gal *et al.* [2], in order to finetune the original StyleGAN [6] generator network (*e.g.*, trained on FFHQ [5] for faces) to produce images from different artistic domains. Different variations in output artistic domain are achieved through simple text commands for source-to-target domain transition (*e.g.*, photo → watercolor painting) without requiring image datasets for the target domain. We then use the encoder-bootstrapping strategy from [1], wherein *paint2pix* encoding is used in conjunction with the domain-adapted generator network in order to infer user-intention in the target domain. The generalizability of *paint2pix* model to different brushstroke variations (or abstractions) is important in this regard, as it allows *paint2pix* to reliably encode image semantics from canvas inputs belonging to different artistic domains (*e.g.*, watercolor paintings) into the latent space of the domain-adapted StyleGAN generator.

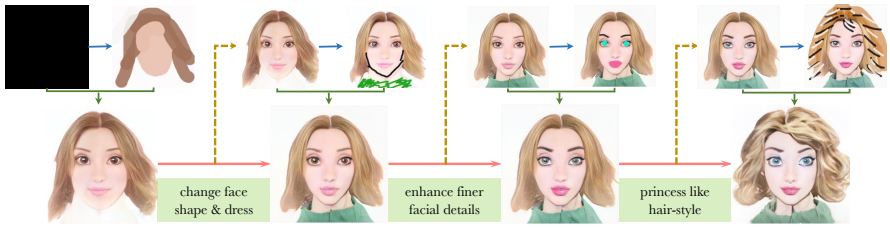
Results. Results are shown in Fig. 1. We observe that similar to photorealistic image synthesis, *paint2pix* allows a novice user to easily express his/her ideas in visual form while progressively synthesizing different styles of high-quality artistic content. Furthermore, we observe that the user-intention predictions from rudimentary user inputs are adapted in order to fit the given artistic context. For instance, coarse scribbles describing the hair of a person are expressed differently for toon (more shiny) and watercolor (more fluid) painting generation. This property is highly convenient, as it implies that a novice user can now create different forms of professional-level art without requiring expertise in the target domain (*e.g.*, oil paintings, colored sketches, *etc.*), through the use of just few generic (not domain dependent) and coarse user scribbles.



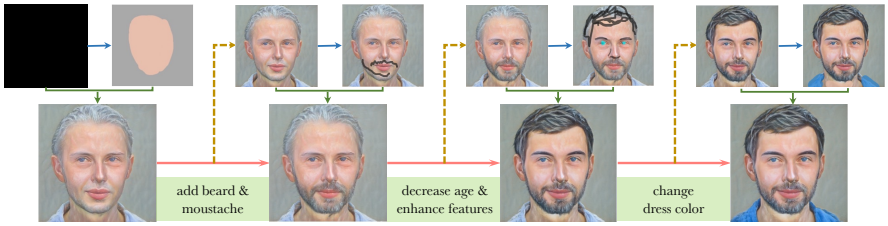
(a) Paint2pix for Metface-like [4] Art Generation



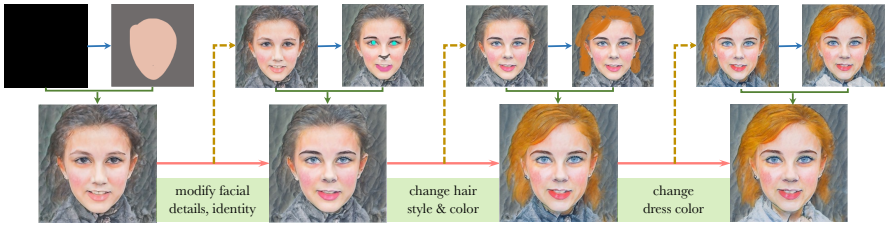
(b) Paint2pix for Colored Sketch Generation



(c) Paint2pix for Disney Cartoon Synthesis



(d) Paint2pix for Oil Painting Synthesis



(e) Paint2pix for Making Watercolor Paintings

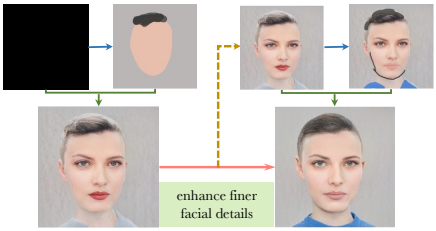
Fig. 1. Analysing paint2pix usage for different forms of artistic content generation.

B Achieving Gender Variation Edits

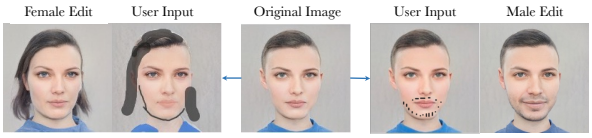
As discussed in the main paper, one of the limitations of *paint2pix* is that it does not provide a direct approach for achieving advanced semantic edits, which are not easily expressed using coarse scribbles. For instance, consider the gender variation edit. A change in underlying gender of a person requires a range of subtle changes in the facial details (*e.g.*, facial hair, hairstyle, eyebrow thickness, use of more masculine/feminine features for face shape, nose, *etc.*). Capturing all these variations using coarse scribbles is not easily feasible. In this section, we describe another way of achieving advanced semantic edits with *paint2pix*, which uses progressive image synthesis (Sec. 4 from main paper) to infer global gender edit direction (Sec. 6 from main paper) in StyleGAN [6] latent space.

Approach. The approach can be summarized as follows,

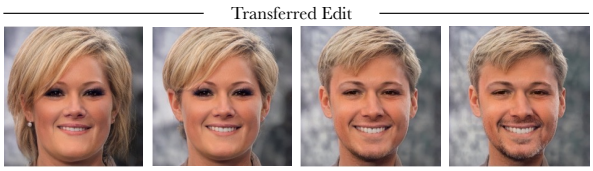
- First, we synthesize a gender neutral hairstyle face. As shown in Fig. 2a, this can be easily achieved with *paint2pix* using just few user scribbles, by starting with off with a gender neutral hairstyle and then selecting the desired output from the set of multi-modal output predictions.
- We next use coarse brushstrokes in order to generate female and male variations from the originally synthesized image (refer Fig. 2b).
- Finally, we use the formulation from Sec. 6 from main paper in order to infer global gender edit direction from the synthesized female (x_0) and male (x_1) facial images. This edit $x_0 \rightarrow x_1$ can then transferred to images across the input domain in order to achieve the gender variation edits (refer Fig. 2c).



(a) Synthesize a gender neutral hairstyle face



(b) Generate male and female variations



(c) Transfer the original edit to new images

Fig. 2. Analysing paint2pix usage for achieving gender variation edits.

C Experiment Details

C.1 Implementation Details

Encoder Design. We use the recently proposed encoder architecture from Alaluf *et al*. [1] for designing both canvas and identity encoders for *paint2pix*. Also we adopt the iterative refinement strategy from [1], wherein starting with a latent code initialization w_t^0, the encoder proceeds to gradually refine the final latent space predictions over N iterative steps. For *paint2pix*, the latent code initialization for the canvas encoder (for both w_t, w_{t+1}) is chosen to be the average StyleGAN [6] latent code for the given input domain. The latent code for the identity encoder is initialized from the latent space prediction w_{t+1} of the canvas encoding stage, which then refines it over N iterative steps to output the final latent prediction $\tilde{w}_{t+1}$. We use $N = 5$ iterative refinement steps for both canvas and identity encoders in our experiments.

Data Augmentation. As discussed in the limitations section (Sec. 9 of the main paper), we note that a key requirement for real image editing is the ability of the used encoder model to invert the original input image onto StyleGAN latent space. For instance, while performing real image editing (*i.e*. initialize C_t as a real image input), the ability of *paint2pix* to perform different semantic edits would depend highly on the ability of the canvas encoder to reconstruct the original input image through output prediction y_t. While the painting annotations used during training include both incomplete and complete paintings, the number of complete paintings used might be insufficient to ensure invertibility of the canvas encoder. In order to ensure additional invertibility of the canvas encoder, we augment the original data to also learn a real-to-real mapping by randomly choosing $C_t = C_{t+1} = \hat{y}_t$ with a 30% probability during training.

C.2 Hyperparameter Summary

A summary of different hyperparamters used during *paint2pix* training is provided in Table 1.

Loss Function	Hyperparameter	Set Value
$\mathcal{L}_{pred}$	λ_1	0.8
	λ_2	0.1
$\mathcal{L}_{edit}$	λ_3	0.4
	λ_4	0.001
$\mathcal{L}_{embed}$	λ_5	0.8
	λ_6	0.001
	λ_7	1.0

Table 1. Hyperparameter selection summary for Paint2pix implementation.

C.3 Quantitative Experiments

In addition to qualitative comparisons, we report quantitative results on the performance of our approach (refer Table 2 of main paper). For each image synthesis / editing task, we report the output image quality using the Fréchet inception distance (FID) [3] between the final output and original input distribution. We also perform a human-evaluation study and report the percentage of human users which prefer our method as opposed to competing works. In this section, we provide further details on the procedure used for performing data collection and human evaluation for the quantitative experiments.

Data collection. Since there is no predefined dataset for custom image synthesis (or edits) using user-scribbles, we create our own dataset for reporting quantitative results. For each image manipulation task (*e.g.*, semantic-image edits), we first use the *paint2pix* user-interface in order to manually generate 200 different $\{C_t, C_{t+1}\}$ tuples, describing a range of possible edits across different images from the CelebA-HQ [7, 8] dataset. The alpha-maps (*rgba* format) from these edits are then transferred to other images from the CelebA-HQ dataset, which share similar facial pose as the original image on which the edit was performed. Finally, the collected edit samples are manually filtered in order to remove possible inconsistencies arising during the edit transfer process (*e.g.*, adding eyebrows on the eye region). The final dataset consists a total of 1k edit samples $\{C_t, C_{t+1}\}$ for each custom image manipulation task discussed in the main paper.

Human user study. While FID [3] scores help evaluate the quality of output predictions, it does not provide information on whether the output image predictions provide a good description of the edit made by the user through coarse scribbles. In order to evaluate the same, we perform a human-evaluation study wherein given the original input image C_t and the coarse user-input in C_{t+1}, the participants are shown different edit outputs (from *paint2pix* and gan-inversion methods) and asked to select the output image which is the most realistic representation of the shown edit. For each task (*e.g.*, semantic-image edits), the collected edit samples (discussed above) were divided among 50 human participants, who were given an unlimited time in order to ensure high quality of the final results. Additionally, in order to remove data noise, we use a repeated comparison (control seed) for each user. Responses of users who answer differently to this repeated seed are discarded while reporting the final results.

References

1. Alaluf, Y., Patashnik, O., Cohen-Or, D.: Restyle: A residual-based stylegan encoder via iterative refinement. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (October 2021) 1, 4
2. Gal, R., Patashnik, O., Maron, H., Chechik, G., Cohen-Or, D.: Stylegan-nada: Clip-guided domain adaptation of image generators. arXiv preprint arXiv:2108.00946 (2021) 1
3. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. Advances in neural information processing systems **30** (2017) 5
4. Karras, T., Aittala, M., Hellsten, J., Laine, S., Lehtinen, J., Aila, T.: Training generative adversarial networks with limited data. Advances in Neural Information Processing Systems **33**, 12104–12114 (2020) 2
5. Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4401–4410 (2019) 1
6. Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., Aila, T.: Analyzing and improving the image quality of stylegan. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8110–8119 (2020) 1, 3, 4
7. Lee, C.H., Liu, Z., Wu, L., Luo, P.: Maskgan: Towards diverse and interactive facial image manipulation. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2020) 5
8. Liu, Z., Luo, P., Wang, X., Tang, X.: Deep learning face attributes in the wild. In: Proceedings of International Conference on Computer Vision (ICCV) (December 2015) 5