Paint2Pix: Interactive Painting based Progressive Image Synthesis and Editing

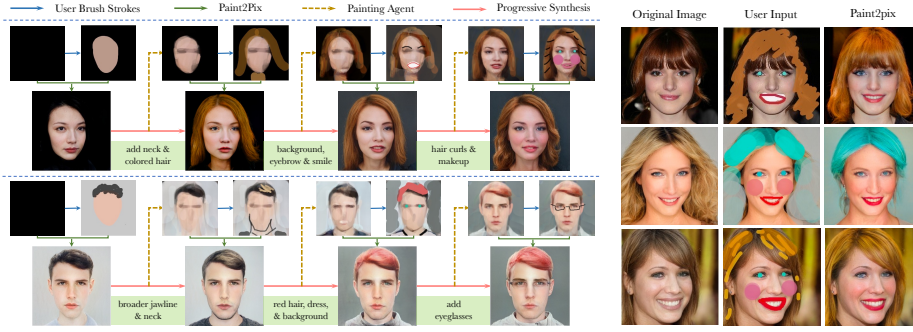
Jaskirat Singh[1,2], Cameron Smith[1], Jose Echevarria[1], Liang Zheng[2],
[1]Adobe Research, [2]Australian National University

Abstract. Controllable image synthesis with user scribbles is a topic of keen interest in the computer vision community. In this paper, for the first time we study the problem of photorealistic image synthesis from incomplete and primitive human paintings. In particular, we propose a novel approach *paint2pix*, which learns to predict (and adapt) "what a user wants to draw" from rudimentary brushstroke inputs, by learning a mapping from the manifold of incomplete human paintings to their realistic renderings. When used in conjunction with recent works in autonomous painting agents, we show that *paint2pix* can be used for progressive image synthesis from scratch. During this process, *paint2pix* allows a novice user to progressively synthesize the desired image output, while requiring just few coarse user scribbles to accurately steer the trajectory of the synthesis process. Furthermore, we find that our approach also forms a surprisingly convenient approach for real image editing, and allows the user to perform a diverse range of custom fine-grained edits through the addition of only a few well-placed brushstrokes.

1 Introduction

The human painting process represents a powerful mechanism for the expression of our inner visualizations. However, accurate depiction of the same is often quite time consuming and limited to those with sufficient artistic skill. Conditional image synthesis provides a popular solution to this problem, and simplifies output image synthesis based on higher-level input modalities (segmentation, sketch) which can be easily expressed using coarse user scribbles. For instance, segmentation based image generation methods [9, 10, 29, 40] allow for control over output image attributes based on user-editable semantic segmentation maps. However, they have obvious disadvantage of requiring large-scale dense semantic segmentation annotations for training, which makes them not easily scalable to new domains. Unsupervised sketch based image synthesis has also been explored [13, 26, 31], but they do not provide control over non-edge image areas.

In this paper, we explore the use of another modality in this direction, by studying the problem of photorealistic image synthesis from *incomplete and primitive human paintings*. This is motivated from the observation that when constrained to a particular domain (*e.g.*, faces), a lot of information about the final image output can be inferred from fairly rudimentary and partially drawn human paintings. We thus propose a novel approach *paint2pix*, which learns to



(a) Progressive image synthesis (b) Real image editing

Fig. 1. Overview. We propose *paint2pix* which helps the user directly express his/her ideas in visual form by learning to predict user-intention from a few rudimentary brushstrokes. The proposed approach can be used for (a) synthesizing a desired image output directly from scratch wherein it allows the user to control the overall synthesis trajectory using just few coarse brushstrokes (blue arrows) at key points, or, (b) performing a diverse range of custom edits directly on real image inputs.

predict (and adapt) "what the user intends to draw" from rudimentary brushstroke inputs, by learning a mapping from the manifold of incomplete human paintings to their realistic renderings. However, learning the manifold of incomplete human paintings is challenging as it would require extensive collection of human painting trajectories for each target domain. For this challenge, we show that a fair approximation of this manifold can still be obtained by using painting trajectories from recent works on autonomous human-like painting agents [5].

While predicting photo-realistic outputs from partially drawn paintings might be helpful for capturing certain parts of a user's visualization (*e.g.*, face shape, hairstyle), fine grain control over different image attributes might be missing. In order to address this need for fine-grained control, we introduce an interactive synthesis strategy, wherein *paint2pix* when used in conjunction with an autonomous painting agent, allows a novice user to progressively synthesize and refine the desired image output using just few rudimentary brushstrokes. The overall image synthesis (refer Fig. 1a) is performed in a progressive fashion wherein *paint2pix* and the autonomous painting agent are used in successive steps. Starting with an empty canvas, the user begins by making few rudimentary strokes (*e.g.*, describing face shape, color) to obtain an initial user-intention prediction (through *paint2pix*). The painting agent then uses this prediction to paint until a user-controlled timestep, at which point, the user again provides a coarse brushstroke input (*e.g.*, describing finer details like hair color) to change the trajectory of the synthesis process. By iterating between these steps till the end of painting trajectory, the human artist is able to gain significant control over final image contents whilst requiring to input only few coarse scribbles (blue arrows in Fig. 1a) at key points of the autonomous painting process.

In addition to progressive image synthesis, the proposed approach can also be used to perform fine-grained editing on real-images (Fig. 1b). As compared with

previous latent space manipulation methods [3,30,32], we find that our approach forms a surprisingly convenient alternative for making a diverse range of custom fine-grained modifications through the use of a few user scribbles. Furthermore, we show that custom edits (*e.g.*, adding smile, changing makeup) are not limited to the image on which the modifications were performed but show generalization across the input domain (Sec. 6). Put another way, once the user is satisfied with a custom edit on one image, the same edit can then be transferred to another image (from the same input domain) in a semantically-consistent manner.

To summarize, the main contributions of this paper are **1)** We introduce a novel task of photorealistic image synthesis from incomplete and primitive human paintings. **2)** We propose *paint2pix* which learns to predict (and adapt) "what a user wants to ultimately draw" from rudimentary brushstroke inputs. **3)** We finally demonstrate the efficacy of our approach for (a) progressively synthesizing an output image from scratch, and, (b) performing a diverse range of custom edits directly on real image inputs.

2 Related Work

Autonomous painting agents. In recent years, substantial research efforts [5, 16, 21, 27, 33, 36, 41] have been focused on developing autonoumous painting agents which can learn an unsupervised stroke decomposition for the recreation of a given target image. Despite their efficacy, previous works in this area are often limited to the non-photorealistic recreation of *a provided target image*. This assumes that the user already has a fixed reference image that he/she wants to recreate. However, in practical applications the intended image output may not be available and has to be synthesized in a progressive fashion. Our work thus proposes to develop a new application for autonomous painting agents by predicting user-intention from incomplete canvas frames.

Segmentation based image generation. Image to image translation frameworks have been extensively studied for controllable generation of highly realistic image outputs based on a more simplified image representation. For instance, [12,17,23,28,29,34,40] use conditional generative adversarial networks for controllable image synthesis using user-provided semantic segmentation maps. While effective, these works require large-scale semantic segmentation annotations for training, which limits their scalability to new domains. Furthermore, making fine-grained changes within each semantic contour after image synthesis is non-trivial and often relies on style encoding methods [10, 40], which require the user to first find a set of reference images which best describe the nature of each intended change (*e.g.* adding makeup or changing hair style for facial images). In contrast, our work allows for a range of custom fine-grained image editions through the addition of just few well placed brush strokes.

Sketch based image generation has also been explored [6,7,13,24–26,37,38]. Ghosh *et al.* [13] predict possible image outputs from rudimentary sketches of simple objects. Richardson *et al.* [31] use an encoder-decoder network to map input sketches to output images in a given domain. While effective in controlling

Method Attribute	*Paint2Pix*	*GAN-Inversion*	*Seg2Photo*	*Sketch2Photo*
From scratch	✓	✗	✓	✓
Responsiveness	✓	✓	✗	✗
No user-expertise	✓	✗	✓	✓
Data efficiency	✓	✓	✗	✓

Table 1. Related work overview. Broad positioning of our work with respect to other methods for controllable image synthesis with user scribbles. (refer Sec. 2 for details)

initial aspects of the image output, the use of sketches (compared to paintings) for image generation is less effective as it offers limited control and sensitivity to changes made in non-edge areas.

GAN inversion. Interactive image generation and editing with user scribbles has also been explored in the context of GAN-inversion methods [1,2,39]. Zhu *et al.* [39] propose a hybrid optimization approach for projecting user-given strokes onto the natural image manifold. Similarly, [1,2] use GAN-inversion to perform local image edits with user scribbles. While effective for small-scale photorealistic manipulations, these methods often lack means to learn the distribution of user-inputs (manifold of rudimentary paintings in our case) and thus are limited to performing a pure color-based optimization. As shown in Sec. 5, this leads to poor performance on from-scratch synthesis and semantic edits on real images.

Positioning our work. Table 1 summarizes the positioning of our approach with respect to previous methods performing controllable image synthesis using user-given brushstrokes/scribbles. In particular, we posit the comparative benefits of our approach with respect to the following desirable properties.

- ***Image synthesis from scratch.*** While Paint2pix, segmentation and sketch based methods allow for direct synthesis of the primary image from scratch, GAN-inversion methods perform a more color-based optimization and thereby show poor performance on image synthesis from scratch (Sec. 5.1).
- ***Responsiveness (control) over all image areas.*** Due to the one-to-many nature of learned mappings, segmentation based methods fail to provide fine-grained control over attributes within each semantic region. Similarly, sketch-based methods lack sensitivity to changes in non-edge areas.
- ***Usability by novice artists.*** A key advantage of our method is that it allows a novice artist to control the synthesis process while using fairly rudimentary brushstrokes. In contrast, GAN-inversion based methods require the user to make sufficiently detailed strokes in order to preserve closeness to the real image manifold (refer Sec. 5.2 for more details).
- ***Data efficiency.*** Our method is completely self-supervised and uses [5] to approximate the manifold of incomplete human paintings. In contrast, segmentation based methods require large-scale dense semantic maps for training on each target domain, which limits their scalability.

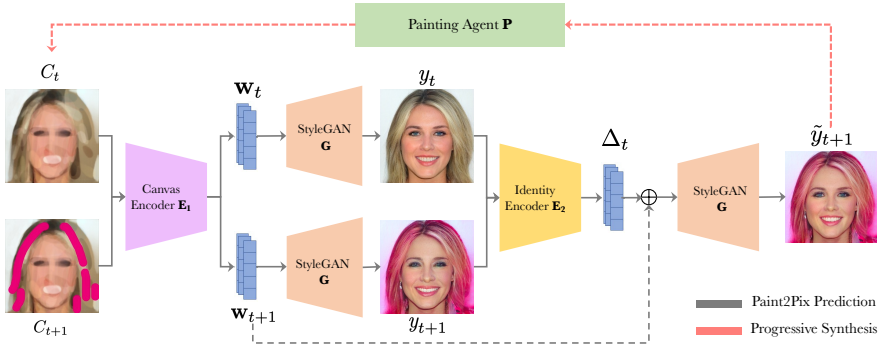


Fig. 2. Model Overview. The *paint2pix* model helps simplify the image synthesis task by predicting user-intention from rudimentary canvas state C_t, while also allowing the user to accurately steer the synthesis trajectory using coarse brushstroke inputs in C_{t+1}. This is done in two steps. First, the canvas encoder $\mathbf{E_1}$ learns a mapping between the manifold of incomplete paintings and real images to predict realistic user-intention predictions $\{y_t, y_{t+1}\}$ from $\{C_t, C_{t+1}\}$ respectively. These intermediate predictions are then fed into a second identity encoder $\mathbf{E_2}$ to predict a latent-space correctional term Δ_t, which ensures that the final prediction $\tilde{y}_{t+1}$ preserves the identity of the prediction from the original canvas C_t, while at the same time incorporating changes made by the user input brushstrokes in C_{t+1}. The progressive synthesis process can then be continued by feeding final prediction $\tilde{y}_{t+1}$ to an autonomous painting agent which paints it till a user-controlled timestep, at which point, the user can again add coarse brushstroke inputs in order to better express her inner ideas in the final image output.

3 Our Method

The *paint2pix* model uses a two step decoupled encoder-decoder architecture (refer Fig. 2) for predicting user intention from incomplete user paintings, while at the same time allowing a novice user to control the overall image synthesis trajectory using coarse and rudimentary brushstroke inputs.

3.1 Canvas Encoding Stage

The goal of the canvas encoding stage is two-fold: 1) predict user-intention by learning a mapping between the manifold of incomplete user paintings to their realistic output renderings, while at the same time 2) allow for modification in the progressive synthesis trajectory based on coarse user-brushstrokes.

In particular, given current canvas state C_t and the updated canvas state after coarse user-brushstroke input C_{t+1}, we first use a canvas encoder $\mathbf{E}_1$ to predict a tuple of initial latent vector predictions $\{\mathbf{w}_t, \mathbf{w}_{t+1}\}$ as,

$$\{\mathbf{w}_t, \mathbf{w}_{t+1}\} = \mathbf{E}_1(C_t, C_{t+1}). \tag{1}$$

These latent predictions are then fed into a StyleGAN [20] decoder network $\mathbf{G}$, in order to get realistic user-intention predictions $\{y_t, y_{t+1}\}$ corresponding to input canvas tuple $\{C_t, C_{t+1}\}$ respectively,

$$\{\mathbf{y}_t, \mathbf{y}_{t+1}\} = \{G(\mathbf{w}_t), G(\mathbf{w}_{t+1})\}. \tag{2}$$

Losses. Given realistic output ground-truth annotation $\hat{y}_t$ corresponding to canvas C_t (refer Sec. 3.3), the canvas encoder $\mathbf{E}_1$ is trained to learn to predict user-intention with the following prediction loss $\mathcal{L}_{pred}$,

$$\mathcal{L}_{pred} = \mathcal{L}_2(y_t, \hat{y}_t) + \lambda_1 \mathcal{L}_{lpips}(y_t, \hat{y}_t) + \lambda_2 \mathcal{L}_{id}(y_t, \hat{y}_t), \tag{3}$$

where $\mathcal{L}_{lpips}$ is the perceptual similarity loss [18] and $\mathcal{L}_{id}$ represents the Arcface [11] / MoCo-v2 [8] features based identity similarity loss from Tov *et al.* [35].

As previously mentioned, we would also like to ensure that the output predictions are modified in order to reflect the changes added by the user in C_{t+1}. This is then achieved by the following edition loss $\mathcal{L}_{edit}$,

$$\mathcal{L}_{edit} = \mathcal{L}_{lpips}(\Delta C_t, \Delta y_t) + \lambda_3 \mathcal{L}_{adv}(w_{t+1}) + \lambda_4 \|w_{t+1} - w_t\|_2, \tag{4}$$

where $\Delta C_t = C_{t+1} - C_t$ and $\Delta y_t = y_{t+1} - y_t$ represent the changes in the original canvas and output predictions respectively. $\mathcal{L}_{adv}$ refers to the latent discriminator loss from e4e [35] to ensure realism of the latent space prediction. Finally, the last term ensures that the codes $\{w_t, w_{t+1}\}$ for consecutive image outputs $\{y_t, y_{t+1}\}$ lie close in the StyleGAN [20] latent space.

The overall loss for the canvas encoding stage is then defined as follows,

$$\mathcal{L}_{canvas} = \mathcal{L}_{pred} + \lambda_{edit}\ \mathcal{L}_{edit}. \tag{5}$$

3.2 Identity Embedding Stage

While enforcing closeness of consecutive latent vector codes $\{w_t, w_{t+1}\}$(Eq. 4) , helps in ensuring that the updated output prediction y_{t+1} is derived from the original prediction y_t, inconsistencies might still arise due to subtle changes in the identity of the underlying prediction (Fig. 2). Thus, the goal of the second stage is to preserve the underlying identity between consecutive image predictions and thereby ensure semantic consistency of the overall image synthesis process.

To address this, we train a second identity encoder $\mathbf{E}_2$ which ensures that the final prediction $\tilde{y}_{t+1}$ preserves identity of the original prediction y_t while still reflecting the changes made by the user in canvas C_{t+1}. In particular, given output image predictions $\{y_t, y_{t+1}\}$ from the canvas encoding stage, the identity encoder $\mathbf{E}_2$ predicts a correctional term Δ_t to update the latent codes as,

$$\tilde{w}_{t+1} = w_{t+1} + \Delta_t, \quad \text{where} \quad \Delta_t = \mathbf{E}_2(y_t, y_{t+1}). \tag{6}$$

The updated latent code $\tilde{w}_{t+1}$ is then used to predict the final output prediction $\tilde{y}_{t+1}$ using the StyleGAN [20] decoder $\mathbf{G}$ as,

$$\tilde{y}_{t+1} = \mathbf{G}(\tilde{w}_{t+1}). \tag{7}$$

Losses. The identity encoder is trained using the following loss,

$$\mathcal{L}_{embed} = \mathcal{L}_2(y_{t+1}, \tilde{y}_{t+1}) + \lambda_5 \mathcal{L}_{lpips}(y_{t+1}, \tilde{y}_{t+1}) + \lambda_6 \|\Delta_t\|_2 + \lambda_7 \mathcal{L}_{id}(y_t, \tilde{y}_{t+1}) \tag{8}$$

where the first three terms ensure the preservation of edits made by the user in C_{t+1}, while the last term enforces that the final prediction $\tilde{y}_{t+1}$ preserves the identity of the original image prediction y_t, thereby ensuring consistency of the overall progressive synthesis process.

Reason for decoupled encoders. While its feasible to design a model architecture wherein both $\mathcal{L}_{canvas}$ and $\mathcal{L}_{embed}$ are applied using a single encoder, the use of a decoupled identity encoder offers several practical advantages. For instance, while ensuring identity consistency is usually important (*e.g.*, making fine-grained changes), a change in underlying identity might sometimes be actually desirable, especially at the beginning of the progressive synthesis process. The decoupling of canvas encoding and identity embedding stage is therefore useful, as it allows the user to apply identity correction depending on the nature of the intended change. Furthermore, as shown in Sec. 7, decoupling the two stages allows our model to perform multi-modal synthesis without requiring any special architecture for producing multiple output predictions.

3.3 Overall Training

Total loss. The overall *paint2pix* model is jointly trained using both canvas-encoding $\mathcal{L}_{canvas}$ and identity embedding $\mathcal{L}_{embed}$ losses:

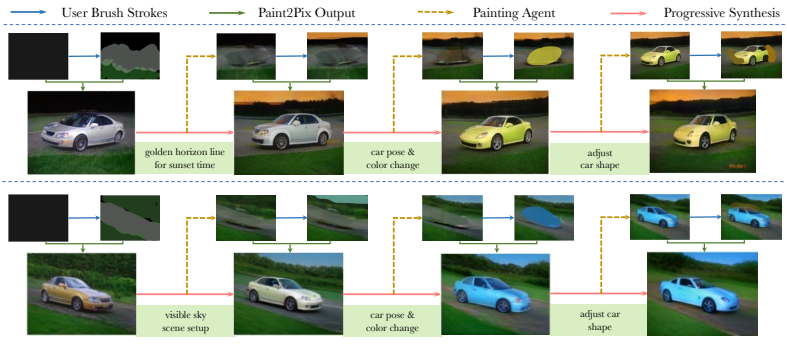
$$\mathcal{L}_{total} = \mathcal{L}_{canvas} + \lambda_{embed}\, \mathcal{L}_{embed}. \tag{9}$$

Ground truth painting annotations. As discussed before, a key requirement of our approach is the ability to learn a mapping between the manifold of incomplete human-user paintings to their ideal realistic outputs. This requirement is challenging as it would need large-scale collection of human painting trajectories for each target domain, making our method intractable for most practical applications. To address this, we propose to instead use the recent works on autonomous painting agents for obtaining a decent approximation for the manifold of incomplete user paintings. The accuracy of such an approximation would depend highly on the domain gap between the incomplete paintings made by human users as compared to those made by a painting agent. We reduce this domain gap by using the recently proposed *Intelli-paint* [5] method, which has been shown to generate intermediate canvas frames which are more intelligible to actual human artists as opposed to previous works [16, 27, 33, 41].

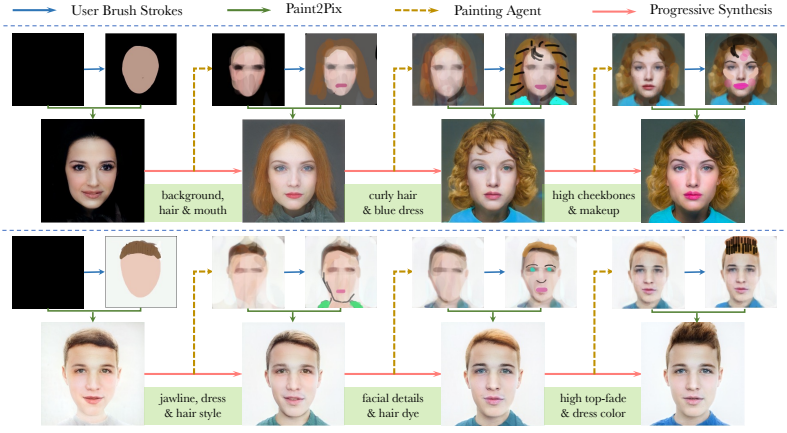
In particular, for each painting trajectory trying to recreate a given target image $I_{target} \in \mathcal{D}$ ($\mathcal{D}$ is input domain, *e.g.*, FFHQ [19] for faces), we collect input canvas annotations by uniformly sampling 20 tuples of consecutive canvas frames $\{C_t, C_{t+1}\}$ observed during the painting process. The output image annotation $\hat{y}_t$ for all sampled canvas tuples (from the same trajectory) is then set to the original target image I_{target}. Furthermore, we collect painting annotations under various brushstroke counts $N_{strokes} \in [200, 500]$, as it helps capture the diverse degrees of abstraction observed in paintings made by actual human artists.

4 Paint2pix for Progressive Image Synthesis

In this section, we demonstrate how the one-step user intention prediction network from Sec. 3 can be used for progressively synthesizing a potential user's ideas as a realistic image output without requiring artist-level expertise.



(a) Progressive Image Synthesis for Cars (Stanford-Cars [22]) domain.



(b) Progressive Image Synthesis for Facial (FFHQ [19]) domain.

Fig. 3. Paint2pix for Progressive Image Synthesis.

Figure 3 demonstrates the use of *paint2pix* for progressive image synthesis from scratch. A potential user would start the painting process by adding a few rudimentary brushstrokes on the canvas (*e.g.*, background scene for cars or face shape, color for faces). The *paint2pix* network then outputs a set of possible realistic image renderings (refer Sec. 7 for more details on multi-modal synthesis) that the user might be interested in drawing. The user may then select the image that most closely resembles his/her idea to obtain a user-intention prediction. The progressive synthesis process can then be continued by feeding this prediction to an autonomous painting agent which paints it till a user-controlled timestep, at which point, the user can again add coarse scribbles (*e.g.*, describing finer details like sky color for cars or hairstyle for faces) in order to steer the synthesis trajectory according to his/her ideas. By continuing this iterative process till the end of the painting process, a novice user can gain significant control over the final image contents while requiring to only input few rudimentary brushstrokes at key points in the autonomous painting trajectory.

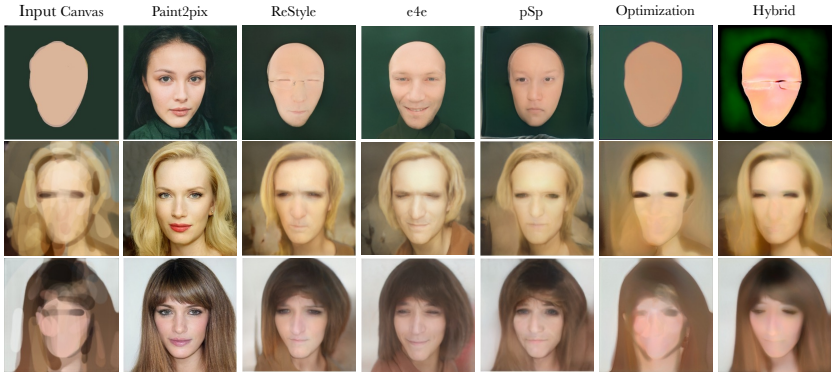


Fig. 4. Qualitative comparison: Predicting user intention from rudimentary paintings.

5 Comparison with Inversion Methods

Interactive image generation and editing with user brushstrokes has also been explored in the context of GAN-inversion methods [1,2,39], which use an encoder or optimization based inversion approach in order to project user scribbles onto the real image manifold. In this section, we present extensive quantitative and qualitative results comparing our approach with existing GAN-inversion methods for image manipulation with user-scribbles. In particular, we demonstrate the efficacy of our approach in terms of both 1) from scratch synthesis: *i.e.*, predicting user intention from fairly rudimentary paintings (Sec. 5.1), and 2) real image editing: allowing a potential user to make a range of custom fine-grain edits directly by just using a few coarse input brushstrokes (Sec. 5.2).

Baselines. We compare our results with recent state-of-the-art encoder based methods from Restyle [4], e4e [35] and pSp [31]. In addition, we report results for *optimization* based encoding approach from Karras *et al.* [20] and *hybrid* strategy from Zhu *et al.* [39]. Please note that in order to get best output quality, results for [39] are reported while using a pretrained ReStyle [4] encoder.

5.1 Predicting User-Intention from Rudimentary Paintings

Qualitative results. Fig. 4 shows qualitative comparisons while predicting photo-realistic (user-intention) outputs from rudimentary paintings. We clearly see that our approach results in much more photorealistic predictions for the user-intended final output. In contrast, the predominantly color-based optimization nature of previous GAN-inversion works leads to non-photorealistic projections when all color details are yet to be added by the user. For instance, while drawing a human face, it is quite common for an artist to first draw a coarse brushstroke for the face region without adding in the finer facial details. However, this leads to poor performance while using color-based optimization as it leads the model to instead predict an output face where the finer facial details are hardly noticeable. Adversarial loss in e4e [31] helps improve the realism of output images but it still performs worse than *paint2pix* for this task.

Quantitative results. We also report quantitative results for this task (and image editing tasks from Sec. 5.2) in Table 2. Results are reported in terms of the Fréchet inception distance (FID) [15], which is used capture the output image quality from different methods. Furthermore, we perform a human user-study (details in supp. material) and report the percentage of human users which prefer our method as opposed to competing works. As shown in Table 2, we observe that *paint2pix* produces better quality images (lower FID scores) and is preferred by majority of human users over competing methods.

5.2 Real-image Editing

In addition to being able to perform progressive synthesis from scratch, *paint2pix* also offers a surprisingly convenient approach for making a diverse range of custom semantic edits (*e.g.*, add smile for faces) on real images by simply initializing the canvas input C_t with a real image. We next compare our method with previous GAN-inversion works on performing real image editing with user scribbles.

Semantic image edits. As shown in Fig. 5, we observe that our approach performs much better when the nature of the underlying edit is not purely color-based. For instance, consider the the first example from Fig. 5. Our method is able to correctly interpret that coarse white brushstrokes near the mouth region implies that the user is trying to add smile to the underlying facial image. In contrast, due to the predominantly color-based-optimization, gan-inversion methods fail to understand the change in semantics of face, and thus predict output faces in which the mouth region has been artificially-colored white.

Fig. 5. Paint2pix for achieving semantic image edits. Best viewed zoomed-in.

Color-based custom edits. Even when the custom-edits are color-based, we show that *paint2pix* leads to outputs which are 1) more photorealistic, 2) exhibit a greater level of detail at the edit locations, 3) modify non-edit locations (in addition to edit locations) in order to maintain coherence of the resulting image and 4) better preserve the identity of the original image input.

Results are shown in Fig. 6. Consider the first example (row-1). The increased realism of *paint2pix* outputs can be clearly seen by the more photorealistic and detailed representation at edit locations (*e.g.*, hair, eyebrows). Furthermore, note

Task	FID (↓) comparison						User Study
	Paint2pix	Restyle	e4e	pSp	Optimization	Hybrid	Paint2pix Preference
From scratch	**40.96**	85.98	79.69	89.83	107.2	91.62	97.32 %
Semantic Edits	**40.24**	45.27	42.32	46.08	47.16	49.29	94.04%
Color Edits	**63.56**	100.2	93.11	107.3	116.4	114.2	93.85%

Table 2. Quantitative evaluation. Col 2-7: FID results for comparing output image quality on different image synthesis, editing tasks. Col-8: Human user-study results reporting percentage of users which prefer Paint2pix outputs over other methods.

that our method shows a more global understanding of image semantics and subtly modifies the skin tone and the eye shading of the face in order to maintain consistency with user-given edits. In contrast, the color-based optimization of GAN-inversion methods exhibit a lower level of detail at edit locations (*e.g.*, hair in row 1-3 and makeup in row 2,3). Furthermore, we find that our method shows better performance in preserving the identity of the original input image in the final output (*e.g.*, row 2,3), which is highly essential for real-image editing.

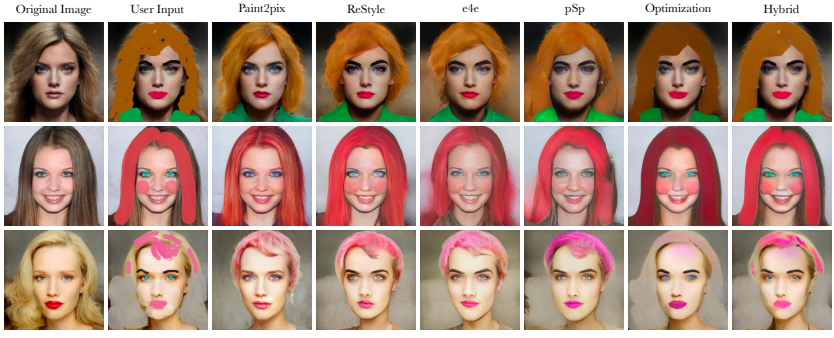


Fig. 6. Paint2pix for achieving color-based custom edits. Best viewed zoomed-in.

6 Inferring Global Edit Directions

We next show that the custom edits (*e.g.*, adding glasses, changing makeup) learned through *paint2pix* are not limited to the image on which the modifications were originally performed but instead show semantically-consistent generalization across the input domain. Put another way, once the user is satisfied with the output of a given custom edit on one image, the same edit can then be applied across different images from the input data distribution without requiring the user to repeat similar brushstrokes on each individual image.

In particular, consider $\{x_0, x_1\}$ be the original and edited image tuple with stylegan latent space vectors $\{\mathbf{w}_0, \mathbf{w}_1\}$ respectively. The custom edit $x_0 \rightarrow x_1$ can then be applied to another image x (with stylegan latent code $\mathbf{w}$) by computing a modified latent space edit direction $\delta_{edit}(x)$ as,

$$\delta_{edit}(x) = \delta_{edit}(x_0) + \mathbf{E}_2(x, \mathbf{G}(\mathbf{w} + \delta_{edit}(x_0))), \tag{10}$$

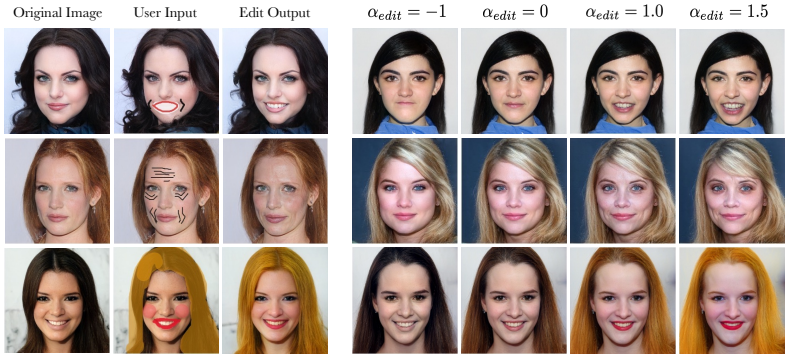


Fig. 7. Inferring global edit directions using Paint2pix. Left: Original custom edit using Paint2pix. Right: Same edit transferred to another image with different edit strengths.

where $\delta_{edit}(x_0) = \mathbf{w}_1 - \mathbf{w}_0$ represents the original edit direction from $x_0 \rightarrow x_1$, and the second term ensures identity preservation in the transferred edit.

The original edit can then be transferred to the input image x as,

$$x' = \mathbf{G}(\mathbf{w} + \alpha\ \delta_{edit}(x)), \tag{11}$$

where α is the edit strength and $\mathbf{G}$ is the styleGAN [20] decoder network.

Results are shown in Fig. 7. We are able to clearly see that custom edits learned on one image can be easily extended to different images in a semantic-consistent manner. Our experiments also reveal that the generalizability of the edit is largely independent of the complexity of the underlying edit. This enables us to transfer both simple (*e.g.*, smile in row-1) and complex edits (*e.g.*, makeup change in row-3) to new images across the input domain.

Furthermore, we observe that the strength of the intended edit can be varied by simply adjusting the edit-strength parameter α. This helps us to use extrapolation in order to achieve edits which would be otherwise difficult to draw using rudimentary brushstrokes. For instance, while adding smile (using white brushstrokes) is easy, drawing a fully laughing face might be difficult for a novice artist. However, the same can be easily achieved by using a higher edit strength which allows us to extrapolate the original smiling edit to a laughing face edit (refer row-1, Fig. 7). Similarly, different levels of facial wrinkles (or aging) can also be achieved in an analogous fashion (refer row-2, Fig. 7).

7 Multi-modal Synthesis

Predicting a single output for inferring user intention from an incomplete painting might not be always useful if the user's ideas are vastly different from the output prediction. The use of decoupled encoders in *paint2pix* is helpful in this regard, as it allows our approach to perform multi-modal synthesis for the final output without requiring special architecture changes.

In practice, given an incomplete canvas C_t, multi-modal synthesis is achieved by sampling a random image as the identity input (y_t) to the identity encoder

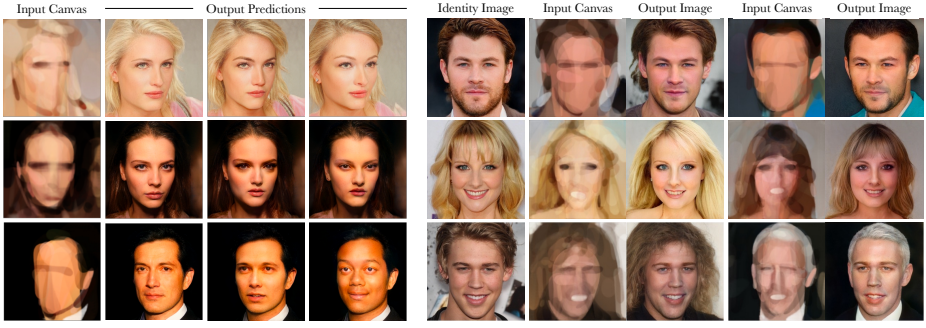


Fig. 8. Paint2pix for (a) multi-modal synthesis and (b) identity-conditioned generation.

network. Results are shown in Fig. 8. We observe that the above approach forms a convenient method for predicting multiple possible image completions from incomplete paintings. This provides the user with a wider range of choices to select the best direction for the synthesis process. Furthermore, we note that this idea can also be used to perform identity conditioned synthesis, by using the same identity image (*e.g.*, Chris Hemsworth) throughout the painting trajectory.

8 Ablation Study

In this section, we perform several ablation studies in order to study the importance of different losses $\{\mathcal{L}_{pred}, \mathcal{L}_{edit}, \mathcal{L}_{embed}\}$ in the performance of *paint2pix*. Please note that in order to still get meaningful results, the experiments without $\mathcal{L}_{pred}$ are performed while using a pretrained restyle [4] network for independently predicting intermediate outputs $\{y_{t+1}, y_t\}$ from canvas frames $\{C_{t+1}, C_t\}$.

Results are shown in Fig. 9. We observe that {w/o $\mathcal{L}_{pred}$} the model lacks an understanding of the manifold of incomplete paintings and thus produces outputs which are not fully photorealistic. In contrast, {w/o $\mathcal{L}_{edit}$} shows high quality outputs but does not incorporate the edits made by user brushstrokes. Finally, we see that the use of $\mathcal{L}_{embed}$ helps the model produce images which preserve the identity of the original image in the final prediction.

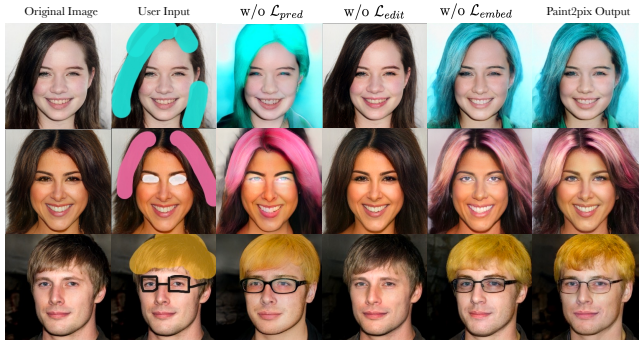


Fig. 9. Ablation study for different losses in Paint2pix. Best viewed zoomed-in.

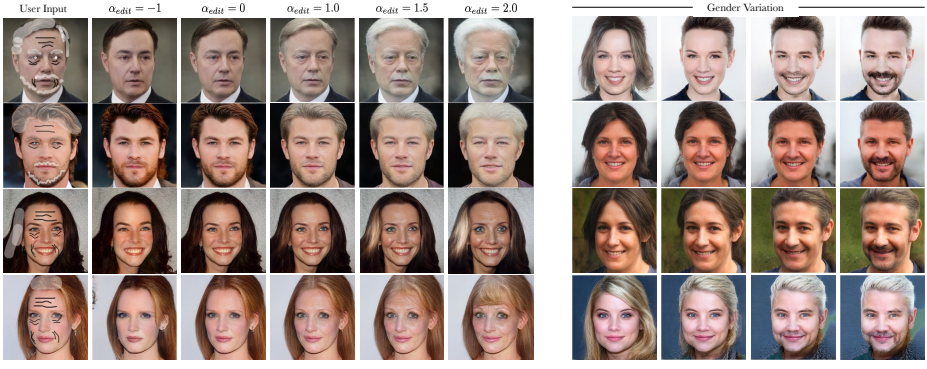
9 Discussion and Limitations

We next provide a discussion of some advanced usage and limitations for *paint2pix*, in order to aid a more holistic understanding of the proposed approach.

In-distribution predictions. A key advantage of *paint2pix* is that allows a novice user to synthesize and manipulate an output image on the real image manifold, while using fairly rudimentary and crude brushstrokes. While this is desirable in most scenarios, it also limits our method as it prevents a potential user from intentionally performing out-of-distribution (or non-realistic) facial manipulations (*e.g.*, blue eyebrows, ghost like faces *etc.*).

Invertibility for real-image editing. Much like other GAN-inversion and latent space manipulation methods [3,4,14,30,31,39], accurate real-image editing with *paint2pix* is highly dependent on the ability of used encoder architecture to invert the original real image into StyleGAN [20] latent space.

Advanced edits. Another limitation is that *paint2pix* does not provide a direct approach for achieving advanced semantic edits like age, gender manipulation. Nevertheless, as show in Fig. 10, age variation edits can still be achieved using extrapolation of edit strength α. Similarly, gender variation edits are possible by using progressive synthesis to infer the gender edit direction. Further details and analysis for gender variation edits are provided in the supp. material.



(a) Age Variation Edits (b) Gender Variation Edits

Fig. 10. Analysing paint2pix usage for achieving advanced semantic edits.

10 Conclusion

In this paper, we explore a novel task of performing photorealistic image synthesis and editing using primitive user paintings and brushstrokes. To this end, we propose *paint2pix* which can be used for 1) progressively synthesizing a desired image output from scratch using just few rudimentary brushstrokes, or, 2) real image editing: wherein it allows a human user to directly perform a range of custom edits without requiring any artistic expertise. As shown through extensive experimentation, we find that *paint2pix* forms a highly convenient and simple approach for directly expressing a potential user's inner ideas in visual form.

References

1. Abdal, R., Qin, Y., Wonka, P.: Image2stylegan: How to embed images into the stylegan latent space? In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4432–4441 (2019) 4, 9
2. Abdal, R., Qin, Y., Wonka, P.: Image2stylegan++: How to edit the embedded images? In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8296–8305 (2020) 4, 9
3. Abdal, R., Zhu, P., Mitra, N.J., Wonka, P.: Styleflow: Attribute-conditioned exploration of stylegan-generated images using conditional continuous normalizing flows. ACM Transactions on Graphics (TOG) **40**(3), 1–21 (2021) 3, 14
4. 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) 9, 13, 14
5. Anonymous, A.: Intelli-paint: Towards developing human-like painting agents 2, 3, 4, 7
6. Chen, T., Cheng, M.M., Tan, P., Shamir, A., Hu, S.M.: Sketch2photo: Internet image montage. ACM transactions on graphics (TOG) **28**(5), 1–10 (2009) 3
7. Chen, W., Hays, J.: Sketchygan: Towards diverse and realistic sketch to image synthesis. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 9416–9425 (2018) 3
8. Chen, X., Fan, H., Girshick, R., He, K.: Improved baselines with momentum contrastive learning. arXiv preprint arXiv:2003.04297 (2020) 6
9. Choi, Y., Choi, M., Kim, M., Ha, J.W., Kim, S., Choo, J.: Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 8789–8797 (2018) 1
10. Choi, Y., Uh, Y., Yoo, J., Ha, J.W.: Stargan v2: Diverse image synthesis for multiple domains. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8188–8197 (2020) 1, 3
11. Deng, J., Guo, J., Xue, N., Zafeiriou, S.: Arcface: Additive angular margin loss for deep face recognition. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4690–4699 (2019) 6
12. Esser, P., Rombach, R., Ommer, B.: Taming transformers for high-resolution image synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12873–12883 (2021) 3
13. Ghosh, A., Zhang, R., Dokania, P.K., Wang, O., Efros, A.A., Torr, P.H., Shechtman, E.: Interactive sketch & fill: Multiclass sketch-to-image translation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1171–1180 (2019) 1, 3
14. Härkönen, E., Hertzmann, A., Lehtinen, J., Paris, S.: Ganspace: Discovering interpretable gan controls. Advances in Neural Information Processing Systems **33**, 9841–9850 (2020) 14
15. 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) 10
16. Huang, Z., Heng, W., Zhou, S.: Learning to paint with model-based deep reinforcement learning. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 8709–8718 (2019) 3, 7

17. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1125–1134 (2017) 3
18. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: European conference on computer vision. pp. 694–711. Springer (2016) 6
19. 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) 7, 8
20. 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) 5, 6, 9, 12, 14
21. Kotovenko, D., Wright, M., Heimbrecht, A., Ommer, B.: Rethinking style transfer: From pixels to parameterized brushstrokes. arXiv preprint arXiv:2103.17185 (2021) 3
22. Krause, J., Stark, M., Deng, J., Fei-Fei, L.: 3d object representations for fine-grained categorization. In: 4th International IEEE Workshop on 3D Representation and Recognition (3dRR-13). Sydney, Australia (2013) 8
23. Lee, C.H., Liu, Z., Wu, L., Luo, P.: Maskgan: Towards diverse and interactive facial image manipulation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5549–5558 (2020) 3
24. Lee, J., Kim, E., Lee, Y., Kim, D., Chang, J., Choo, J.: Reference-based sketch image colorization using augmented-self reference and dense semantic correspondence. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5801–5810 (2020) 3
25. Li, X., Zhang, B., Liao, J., Sander, P.V.: Deep sketch-guided cartoon video synthesis. CoRR (2020) 3
26. Liu, R., Yu, Q., Yu, S.X.: Unsupervised sketch to photo synthesis. In: European Conference on Computer Vision. pp. 36–52. Springer (2020) 1, 3
27. Liu, S., Lin, T., He, D., Li, F., Deng, R., Li, X., Ding, E., Wang, H.: Paint transformer: Feed forward neural painting with stroke prediction. arXiv preprint arXiv:2108.03798 (2021) 3, 7
28. Liu, X., Yin, G., Shao, J., Wang, X., et al.: Learning to predict layout-to-image conditional convolutions for semantic image synthesis. Advances in Neural Information Processing Systems **32** (2019) 3
29. Park, T., Liu, M.Y., Wang, T.C., Zhu, J.Y.: Semantic image synthesis with spatially-adaptive normalization. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2337–2346 (2019) 1, 3
30. Patashnik, O., Wu, Z., Shechtman, E., Cohen-Or, D., Lischinski, D.: Styleclip: Text-driven manipulation of stylegan imagery. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2085–2094 (2021) 3, 14
31. Richardson, E., Alaluf, Y., Patashnik, O., Nitzan, Y., Azar, Y., Shapiro, S., Cohen-Or, D.: Encoding in style: a stylegan encoder for image-to-image translation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2021) 1, 3, 9, 14
32. Shen, Y., Zhou, B.: Closed-form factorization of latent semantics in gans. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1532–1540 (2021) 3

33. Singh, J., Zheng, L.: Combining semantic guidance and deep reinforcement learning for generating human level paintings. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (2021) 3, 7
34. Sushko, V., Schönfeld, E., Zhang, D., Gall, J., Schiele, B., Khoreva, A.: You only need adversarial supervision for semantic image synthesis. arXiv preprint arXiv:2012.04781 (2020) 3
35. Tov, O., Alaluf, Y., Nitzan, Y., Patashnik, O., Cohen-Or, D.: Designing an encoder for stylegan image manipulation. arXiv preprint arXiv:2102.02766 (2021) 6, 9
36. Wang, Q., Guo, C., Dai, H.N., Li, P.: Self-stylized neural painter. In: SIGGRAPH Asia 2021 Posters, pp. 1–2 (2021) 3
37. Xiang, X., Liu, D., Yang, X., Zhu, Y., Shen, X., Allebach, J.P.: Adversarial open domain adaptation for sketch-to-photo synthesis. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 1434–1444 (2022) 3
38. Yang, S., Wang, Z., Liu, J., Guo, Z.: Controllable sketch-to-image translation for robust face synthesis. IEEE Transactions on Image Processing **30**, 8797–8810 (2021) 3
39. Zhu, J.Y., Krähenbühl, P., Shechtman, E., Efros, A.A.: Generative visual manipulation on the natural image manifold. In: European conference on computer vision. pp. 597–613. Springer (2016) 4, 9, 14
40. Zhu, P., Abdal, R., Qin, Y., Wonka, P.: Sean: Image synthesis with semantic region-adaptive normalization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5104–5113 (2020) 1, 3
41. Zou, Z., Shi, T., Qiu, S., Yuan, Y., Shi, Z.: Stylized neural painting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15689–15698 (2021) 3, 7