SaveFace: Controlling Face in Image Diffusion Models

Ashna Khetan, Laya Iyer, Isabel Sieh
Stanford University
Palo Alto, CA
{ashnak, laya, isabelrs}@stanford.edu

Abstract

We present SaveFace, a text-image diffusion model with a face image condition, that takes an input of a face image and an optional text prompt, and generates an image that maintains the face from the original image whilst still changing style.

Text-to-image diffusion models, such as Stable Diffusion, have performed remarkably on image generation tasks, even on faces. However, these large image-generation models tend to contain social biases that harm underrepresented genders and ethnicities. ControlNet introduces conditional inputs such as edge maps into large pre-trained diffusion models for users to have control over the images generated. Current uses of ControlNet can preserve edges or body-pose but tend to homogenize facial features and lose unique facial features such as skin tone, thus indirectly changing the original ethnicity and gender.

To address this, we propose SaveFace in order to keep a face consistent while leveraging the creative flexibility of large text-to-image diffusion models. Our approach combines ControlNet's architecture of Stable Diffusion, facial landmark prediction, and image pixel pre-processing. The contribution of SaveFace is two fold: firstly our research emphasizes diverse societal representation in image diffusion models, secondly SaveFace improves space of diffusion based graphic design that combines real, specifically real faces, and generated images.

1. Introduction

Uncovering the hidden facets of diffusion models and how they operate, allows us to unleash the creative potential of AI-assistants in graphic related fields. Having adequate resources and time to design promotional materials poses a bottleneck for event planners, companies, and small organizations. Generating graphics and marketing materials with appropriate and customized content is not an easy feat; current diffusion models manipulate input images and texts due to the nature of the model itself [23].

However, as we harness the power of these models, it is our responsibility to make sure that we are doing so in an ethical and unbiased manner. A challenge with allowing diffusion models free reign over creativity is the inadvertent erasure of racial and facial features. Part of the reason for this erasure is due to training data of most diffusion models, but also because of the lack of models that allow us to control these features when applying diffusion to an image. Instances of this bias have surfaced across various platforms, most notably with the Bridgerton filter on TikTok, which has been criticized for 'whitewashing' users by altering their appearance to fit Eurocentric beauty standards [2].

Diffusion models have a tendency to homogenize features which can be harmful to those who want to preserve these features in their creatively crafted diffusion outputs. ControlNet [31] has been introduced as a model that allows for the isolation of features in an image which are frozen in the layers of the model and preserved in the final output image. Although vanilla ControlNet does demonstrate an ability to keep image features constant, this doesn't extend to facial features and race.

Throughout this paper we discuss our model SaveFace, which is a further trained version of ControlNet, created specifically to freeze facial features and race of a human face in an image.

2. Related Works

2.1. Image Generation

There have been considerable advances image generation, including Generative Adversarial Networks (GANs) [9] which rely on adversarial training to motivate an image generation task, Variational Autoencoders (VAEs) [16, 5] which learn encodes input data into a lower-dimensional space and decodes it to generate similar images, and most recently Diffusion which iteratively denoises data to generate new images [6, 26]. Diffusion models offer improved sample quality and stability [6], avoiding issues like model collapse seen in GANs and blurry outputs from VAEs. Improvements to diffusion include Latent Diffusion Models [23], which perform the denoising process in the latent space rather than high-dimensional space to make the process more efficient, and image-text models like CLIP [21] that create embedding between image and text allowing for text-to-image diffusion. There also exists image-to-image diffusion models, such as SeeCoder[30, 24], but the most popular state-of-the-art image large-scale diffusion models, such as Stable Diffusion (SD) [23] and DALL-E2 [20], focus on text-to-image generation. GANs [15] and image diffusion models [17] are able to produce faces at a photo-realistic level.

2.2. Task-Specific or Controlled Image Generation

Task-specific image generation, such as style transfer – following a brushstroke style or color scheme from another image– is achieved through older models like Convolutional Neural Networks [8] or CycleGAN [33]. Diffusion models can perform style transfer and other tasks like image inpainting, colorization, and uncropping [24, 23]. However, in the context of our research goal, style transfer methods are limited in their ability to retain the facial details of the original image, while image inpainting for background modification techniques are unable to simultaneously style a person. There is also a rise in prompt engineering text-to-image large diffusion models like DALL-E2 [20], but these approaches require a lot of human description and cycles of re-editing a prompt. Moreover, commercial text-to-image diffusion models that take in an image like ChatGPT [19]with DALL-E2 [20] are creative but unable to retain the face of the original image and reconstruct a homogenized design instead.

InstructPix2Pix [3] and ControlNet [31] are two models that add control to text-to-image diffusion models. IntructPix2Pix is a conditional diffusion model that combines GPT-3 [4] and SD in order to edit images based on human text intructions. ControlNet is also designed to learn conditional controls for pretrained text-to-image diffusion models. Their work acknowledges that large text-to-image diffusion models like SD [23] are trained on billions of images. Thus, finetuning for conditional controls for these models is challenging due to limited data for specific conditions, which can lead to overfitting or catastrophic forgetting. ControlNet adresses this problem. It essentially locks the parameters of the large model, while allowing the encoding models to be trained in order to learn diverse conditional controls. ControlNet's design therefore leverages the strengths of a pretrained model while adapting to new tasks with limited data. It can control SD with conditioning inputs including edges, scribbles, segmentation masks, and depths with and without accompanying text prompts. Notably to our research, ControlNet is able to control image generation with Canny edges and human pose, suggesting the possibility of controlling the face in a similar manner through edges and human features. ControlNetMediaPipeFace [27] maintains face pose, but does not retain the original image's identity. Similarly, Canny edges with ControlNet are successful in largely maintaining the face post and general features of a face, but can drastically change aspects like ethnic features – skin color, dimensions of facial features, hairstyle – or gender.

2.3. Mitigating Social Bias in Diffusion

Despite the high performance of image diffusion models, it still exhibits social biases particularly in relation to gender and ethnicity [18]. For instance, when explicitly prompted to depict various combinations of gender and ethnicity, the range of visual features were expected, but in a more standard use case where social attributes are left unspecified, SD and Dall-E2 are much less diverse. The SD representation of identities surrounding "White/Caucasian/Man" was around 50% while identities surrounding "Black/African American/Woman" was around 4%. This indicates that social attributes may be under-represented. This suggests that in the context ControlNet, the ethnicity of a face produced is more likely to be white, as shown in Figure 1.

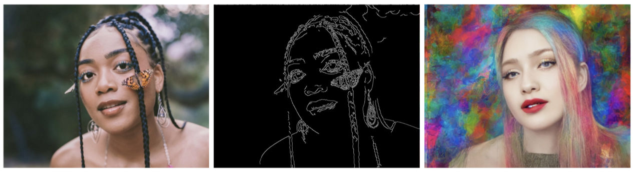

Figure 1. Input, Canny edge condition, and output image

A few approaches have recently attempted to address this at a large scale through distributional alignment and fine tuning text-to-image diffusion models [25], through text guidance [11]. Another approach adjusts the latent code of data to produce unbiased results without needing to fine-tune the model itself [14].

2.4. Facial Landmarks

Lastly, relevant to this res to this research are how images of face data can be represented. Some tools to estimate facial landmarks and depth from a 2d image of a face include OpenCV2's facial landmark [28], and zero-shot depth estimation [22], however, these are limited in visual detail and focus on the pose of the face rather than more detailed characterstics. MediaPipe's Face Landmarker [1] outputs 478 3-dimensional face landmarks and provides a visual mesh representing this data. It's important to note that ControlNetMediaPipeFace[27] only uses high level points that mark the eyebrows, mouth, and eyes and does not take full advantage of facial characterstics that MediaPipe's Face Landmarker can provide.

These relevant works suggest the need for a conditional image diffusion model that controls the face, and hence attributes like ethnicity, of a given image through new facial landmark estimation.

3. Methodology

Our goal is to diffuse a person's environment, background, accessories, etc while maintaining one thing constant: their face. We considered approaches of overlaying an alpha mask over the face, diffusing the background, and inpainting the face onto the generated image. However, we would then have to apply style transfer onto the face as well. Through a learned process, we can avoid such a discretized process.

3.1. ControlNet

We use ControlNet, a neural network architecture, to control a human's facial features and tones while diffusing everything else around it. Controlnet reuses the encoding layers and pretraining from a given diffusion model and learns a specific set of conditional controls. . It maintains a trainable and a locked copy of the diffusion model's parameters. While the locked copy maintains the pretraining learnings, the trainable copy learns parameters specific to the training set given during ControlNet training. The layers are connected with a zero convolution layer (a 1x1 convolutional layer with weights and biases initialized to zeros) which grows with the training process in order to manage datasets of different scales, leading to high performance on smaller and larger datasets. Zero-initializeation ensure the model doesn't start off propogating strong signals from initial examples, allowing it to be more sparse and picky with what patterns are important. ControlNet also does not add noise during training. Since the locked layer already contains learnings, these features combined allow us to focus on the specificity of the control. [explain zero conv math]

The complete ControlNet then computes

$$y_c = F(x;\Theta) + Z(F(x + Z(c;\Theta_{z1});\Theta_c);\Theta_{z2}) \qquad (1)$$

where y_c is the output of the ControlNet block, Z is the zero-convolution layer, $F(x;\Theta)$ is a trained neural block, and c is the condition. In the first training step, since both the weight and bias parameters of a zero convolution layer are initialized to zero, both of the Z($\cdot$; $\cdot$) terms in Equation (1) evaluate to zero, and $y_c = y$, hence the no noise added.

Similar to the original ControlNet, we use Stable Diffusion as our model, though any is replaceable. Stable Diffusion is a U-Net with 12 encoder and decoder blocks, which are convolutional layers, ResNet layers, and Vision Transformers (ViT). Text prompts are encoded with CLIP and position is encoded as diffusion timesteps.

ControlNet applies the control structure to each of the 25 neural blocks. The ControlNet loss is as follows:

$$L = \mathbb{E}_{z_0,t,c_t,c_f,\epsilon\sim\mathcal{N}(0,1)}\left[\|\epsilon - \epsilon_\theta(z_t,t,c_t,c_f)\|_2^2\right] \qquad (2)$$

Stable Diffusion models denoise images in latent space rather than in pixel space. This means that images are converted from 512x512 pixel-space images into several smaller 64x64 latent images using four convolutional layers (kernel size 4, stride 2) with ReLU activation layers. This means that we will do the same with our conditioning images. We chose a kernel size of 4 to balance specificity with efficiency, and given our focus on facial features, the 64-pixel dimension works quite well.

3.2. Face Controls

We created 4 different condition masks to experiment with **Facial Landmarks (Mesh), Facial Landmarks + AlphaMask, Facial Landmarks + Gaussian Blur, and Facial Landmarks + Single-Color** (see Figure 2)

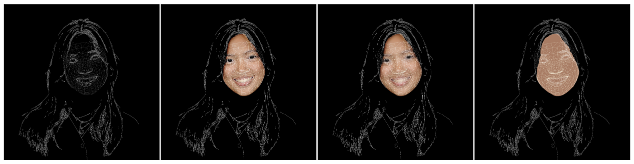

Figure 2. Mesh, AlphaMask, Gaussian Blur, Single-Color

The original model wished to control an images edges, and depth field and a given subject's pose, so they came with the following condition controls: Canny Edge[CITE], Depth Map, Normal Map, M-LSD Lines, HED soft edge, ADE20K segmentation, Openpose, and user sketches. Though these do a great job of capturing an image's overall edges and depth and patterns across the entire viewport, we wished to focus specifically on the face of a person in the image. Specicially, in order to control their facial lines, shadings, and tone, we created the following condition masks.

All condition masks contain the same white outline built off the Canny edge outline, and a visualization of MediaPipe Face Landmarker's Face mesh model. The Face mesh model is an estimate of 478 3-dimensional face landmarks. This combination builds on top of the success of the Canny edge condition masks, which is able to get a rough outline of a face, but enhances the mask through the Face mesh model which not only gets a more intricate face outline but also visualizes depth which was a weakness of control from our past observations. Although the standard MediaPipe Face Landmarker also has thick outlines for eyebrows, eyes, and mouth, we chose not to include these as it is already included in the Face mesh, and could make the face more generalized rather than accurate to the original image.

Mesh refers to only the white outline. The latter 3 condition masks refer to the way in which a detected face is filled with color. This control was to address the change of skin color and tone seen. MediaPipe Face Landmarker limits itself to distinct facial features rather than the entire segment of a face, so we use ImageSegmenter to segment a person's entire face. AlphaMask fills the face segment with the face from the original image, Single-Color takes the average mean of pixels within the face segment, and Gaussian-Blur uses a gaussian blur of face segment from the original image. For these 3 masks, the white outline is placed on top of the face segment.

3.3. Evaluation Metrics

We evaluate our model against 3 models that perform text-image generation conditioned on an image: Control Net Canny, ControlNetMediaPipeFace, and Pix2PixImg2Img.

3.3.1 Quantitative

For our quantitative metrics, we focus on two main image comparison techniques: SSIM (Structural Similarity Index [29]) and PSNR (Peak Signal-to-Noise Ratio [12]). When calculated, the Structural Similarity Index value ranges between -1 and 1. It tells us how structurally similar two images are (higher means more structurally similar). The Peak Signal-to-Noise Ratio, on the other hand, does not have a range, but higher values mean that the images are of better quality. Typically, a PSNR value above 30 dB is considered good quality for most image processing tasks.

We use a dataset of 180 unseen images passed through Control Net Canny, ControlNetMediaPipeFace, Pix2PixImg2Img, and our model and compare the SSIM and PSNR values of each of the generated images from the model with the original image.

3.3.2 Qualitative

For qualitative metrics, we used Average Human Ranking metric modified from ControlNet's user study [31], we sample 22 unseen face portraits paired with a caption and assign each photo to 3 methods: canny, media pipe , and our . We then invited 12 users to rank these 22 groups of 3 results in terms of "the quality of the displayed image" and "the fidelity to the face". This gives us 66 rankings per user, 792 in total, for both result quality and face fidelity. Then we use Average Human Ranking as a preference metric where users rank each result on a scale of 1 to (lower is worse).

The face portraits chosen for evaluation varied in age, sex, ethnicity, and style of photo. The ages range from around 1 year old to 80 years old, the sex has a 12/10 female/male split, and the skin tones range from Tone 2 to 9 on the Monk Skin Tone scale. The style of photo ranges from a professional sit down shoot to athletes mid-game, and contain occluding objects like holding signs or other objects, wearing unconventional hats, having a hand held up, and varied shadows on the face. The captions were chosen after a variety of experiments on the Canny baseline. The captions themselves were generated using ChatGPT 4o for creative variety (see Appendix for prompt). We tested long descriptive prompts (e.g.), short prompts (e.g.), and prompts that specify the image quality of the person (Figure shows some results for each). We found that the prompt with the template "" was the best in rendering a visually appealing background and high quality face that both blended with the theme of the overall photo but attempted to maintain features from the original image.

4. Dataset

We know that ControlNet performs well on datasets of size 50k, so in order to emulate that, our dataset was a 54k mixed bag of several different sources.

- **UTKFaces (20k)** - labeled with age, gender, ethnicity and varies in pose, facial expression, illumination, occlusion, and resolution. [32]
- **Labeled Faces in the Wild (LFW) (13k)** - deep-funneled (refined to learn facial feature representations) images meant to aid with facial recognition tasks. [13]
- **Human Faces (Kaggle) (7k)** - web-scraped human faces across common creeds, races, age groups and profiles, with a few AI-generated sprinkled in [10]
- **10k AI-Generated (10k)** - fully-generated facial images. [7]

Only UTKFaces contains annotations. In ControlNet's implementation, they randomly turned 50% of the captions off (replaced with empty string), so this split emulates that.

Of the 54k, we set aside 5k for testing and 5k for validation.

4.1. Pre-Processing

As seen in the results from Canny-ControlNet, the resulting image often generates multiple distorted faces, or other odd artifacts. This is because SD is trained on images of 512x512 or 512x768, so when faced with images of larger dimension, have difficulty discerning how to fill the extra space.

Therefore, images were first resized to 512x512, as in the paper. Then, the aforementioned filters were applied to generate the four condition images per image. In order to maintain consistency and use the cv2 library for masking, we converted all condition images (RGB, RGBA, etc) to the BGR format.

Finally, UTKFace embedded its annotations in its filename (e.g.[age]_[gender]_[race]_[datetime].jpg where [age] is an integer from 0 to 116, indicating the age, [gender] is either 0 (male) or 1 (female), [race] is an integer from 0 to 4, denoting White, Black, Asian, Indian, and Others (like Hispanic, Latino, Middle Eastern), and [datetime] is in the format of yyyymmddHHMMSSFFF, showing the date and time an image was collected to UTKFace. We converted these to textual descriptions matching ”a photo of a [age]-year-old [ethnicity] [gender]”.

The data’s final form is four sets of triplets of original resized image, condition resized image, and text prompt, one for each mask type.

5. Experiments

We ran for 8500 steps with a learning rate of 1e-5, and a batch size of 1. The original paper found fast training after 4000 steps with a rate of 1e-5 and batch size 4 in 50 minutes; we reduced the batch size to fit memory constraints. We also use a number of methods to save memory: We also used a mixed precision of fp16, which uses 16 bits to represent floating-point numbers, thereby reducing memory footprint and increasing computational throughput. We accumulate gradient for four batches (steps) before updating it. We use gradient checkpointing, which selectively recomputes some activations during the backwards pass rather than caching all variables. Finally, we use the Adam optimizer in order to use 8bit-adam, which converts values to 8-bit precision before computing optimization, use xformers for memory-efficient attention, and set gradients to none after every optimization step. We were able to run for 8.5k optimization steps.

5.1. Baseline

As mentioned earlier we have 3 baselines with which we use to ground our model and deeply understand its results: ControlNet-Canny, ControlNet-MediaPipeFace, and InstructPix2Pix.

6. Results + Discussion

We compare SaveFace to 3 models: ControlNet-Canny, because it preserves edges that could be seen in a face, ControlNet-MediaPipeFace because it also used facial landmarks, and InstructPix2Pix, because it focuses heavily on the instruction given, so we could determine the need for a strong prompt. See results in Figure 3 and 4

6.1. Quantitative

Our SSIM and PSNR scores are as seen in Figure 5. Our model produced results that, by these measures, as least similar to the input image, with InstructPix2Pix performing highest in similarity. We speak in Limitations on causes for this low metric.

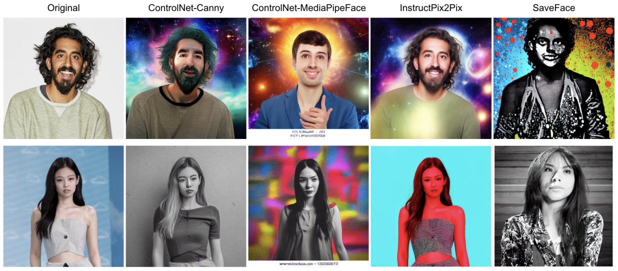

Figure 3. Results from diffusion.

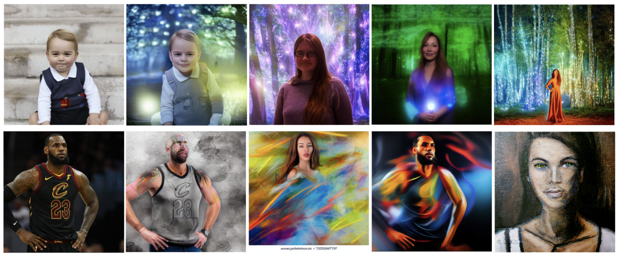

Figure 4. More results from diffusion.

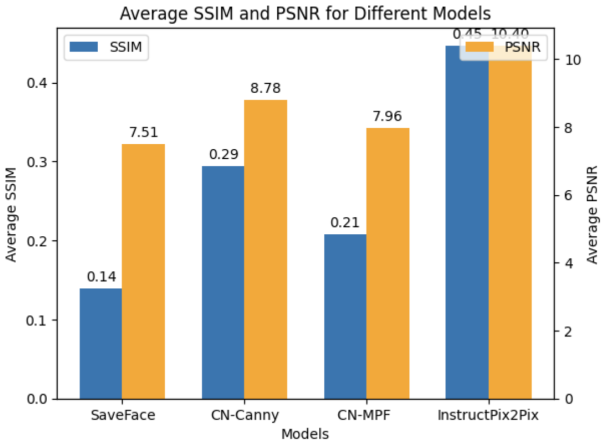

Figure 5. Quantitative Evaluation of Models

6.2. Qualitative

Method	Result Quality	Face Fidelity
ControlNet-Canny	1.8	2.8
ControlNet-MediaPipeFace	2.6	3.2
InstructPix2Pix	2.3	1.3
SaveFace	**3.4**	**3.1**

Participants pointed out that the facial pose, made up of the human’s angle, the way they look at the camera, etc, was captured well by our model. This shows that the mesh landmarks provided valuable information to the model, perhaps just not *enough* valuable information. They also noted that the background imagery and overall diffusive nature followed the prompt quite well, sometimes overpowering and changing the person in the image as well.

7. Conclusion

Qualitatively, our model performed well at maintaining a person's facial pose, though it left other characteristics up to the diffusion. We note creative backgrounds and see much potential for grounding faces, given more work.

7.1. Limitations

Diffusion models undergo the Sudden Convergence Phenomenon, where the model generates concepts quite distant from the original image for many steps and then suddenly learns the input condition (see Figure 6). This occurs around 10k steps normally, and 6k in some of the examples from Canny-ControlNet. Knowing this convergence point requires experimentation. Due to memory and time constraints, we could only keep one checkpoint at a time and could not compare the model at any more than two stages during training. It is possible that, given more epochs, it would converge to something that looks closer to the initial image.

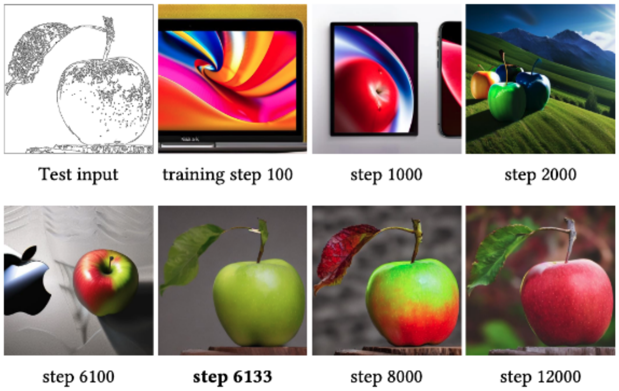

Figure 6. Sudden Convergence Phenomenon

We also ran evaluation on a limited amount of data, whereas we may see larger patterns if we ran on 1k+ images.

We also lacked a clear quantitative evaluation metric that promotes creativity but also ensures that the faces remain constant. Open-source generative models are not yet good at assessing images with such attention to detail and there doesn't exist a proper benchmark that doesn't use such methodology.

7.2. Future Work

We had several experiments we would run given more time and resources. We would vary the mask type, the base model, and the caption type given in training. More specifically, we wonder if our ControlNet would perform better if trained on top of another ControlNet, perhaps depth, so that we maintain the depth of facial features as well as their structure. Preliminary tests showed that combining Canny with any version of a mesh mask would not bode well, since it would take the mesh lines as literal edges. We also wished to experiment with captions that described the individual's face in more detail during training (e.g. ”Older Asian woman with straight black hair, a sharp nose, and wrinkles on her forehead. She has pink lips...”), which would help the model to focus on specifics in the image but could also lead to over-reliance on the prompt.

8. Contributions

Ashna loaded the data, trained the model, and wrote Methodology, Data, and Experiments. Isabel worked on qualitative eval, the condition image, and wrote the related works and made the figures. Laya worked on quantitative eval and wrote Introduction and Quantitative.

References

[1] G. AI. Mediapipe face landmarker. https://ai.google.dev/edge/mediapipe/solutions/vision/face_landmarker, 2023. Accessed: 2023-06-06. 2

[2] G. Baker-Whitelaw. Tiktok's ai outpainting filter accused of whitewashing images. https://www.dailydot.com/irl/tiktok-ai-outpainting-filter-whitewashing/, 2023. Accessed: 2023-06-06. 1

[3] T. Brooks, A. Holynski, and A. A. Efros. Instructpix2pix: Learning to follow image editing instructions, 2023. 2

[4] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei. Language models are few-shot learners, 2020. 2

[5] L. Cai, H. Gao, and S. Ji. Multi-stage variational auto-encoders for coarse-to-fine image generation, 2017. 1

[6] P. Dhariwal and A. Nichol. Diffusion models beat gans on image synthesis, 2021. 1

[7] R. Erkhov. Ai-generated faces. https://github.com/RichardErkhov/ai_generated_faces, 2023. Accessed: 2023-06-06. 4

[8] L. A. Gatys, A. S. Ecker, and M. Bethge. A neural algorithm of artistic style, 2015. 2

[9] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial networks, 2014. 1

[10] A. Gupta. Human faces (kaggle). https://www.kaggle.com/datasets/ashwingupta3012/human-faces?resource=download-directory, 2022. Accessed: 2023-06-06. 4

[11] R. He, C. Xue, H. Tan, W. Zhang, Y. Yu, S. Bai, and X. Qi. Debiasing text-to-image diffusion models, 2024. 2

[12] A. Horé and D. Ziou. Image quality metrics: Psnr vs. ssim. In *2010 20th International Conference on Pattern Recognition*, pages 2366–2369, 2010. 4

[13] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller. Labeled faces in the wild (lfw). https://www.kaggle.com/datasets/jessicali9530/lfw-dataset?select=pairs.csv, 2007. Accessed: 2023-06-06. 4

[14] Y. Jiang, Y. Lyu, T. Ma, B. Peng, and J. Dong. Rs-corrector: Correcting the racial stereotypes in latent diffusion models, 2023. 2

[15] T. Karras, S. Laine, and T. Aila. A style-based generator architecture for generative adversarial networks, 2019. 2

[16] D. P. Kingma and M. Welling. Auto-encoding variational bayes, 2022. 1

[17] L. Lin, Santosh, X. Wang, and S. Hu. Ai-face: A million-scale demographically annotated ai-generated face dataset and fairness benchmark, 2024. 2

[18] A. S. Luccioni, C. Akiki, M. Mitchell, and Y. Jernite. Stable bias: Analyzing societal representations in diffusion models, 2023. 2

[19] OpenAI. Chatgpt: A conversational agent based on gpt-4. https://www.openai.com/chatgpt, 2023. Accessed: 2024-06-05. 2

[20] OpenAI. DALL-E 2. https://openai.com/product/dall-e-2, 2023. 1, 3. 2

[21] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever. Learning transferable visual models from natural language supervision, 2021. 2

[22] N. Rogge. Dpt depth estimation. https://huggingface.co/spaces/nielsr/dpt-depth-estimation, 2023. 2

[23] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. High-resolution image synthesis with latent diffusion models, 2022. 1, 2

[24] C. Saharia, W. Chan, H. Chang, C. A. Lee, J. Ho, T. Salimans, D. J. Fleet, and M. Norouzi. Palette: Image-to-image diffusion models, 2022. 2

[25] X. Shen, C. Du, T. Pang, M. Lin, Y. Wong, and M. Kankanhalli. Finetuning text-to-image diffusion models for fairness, 2024. 2

[26] J. Sohl-Dickstein, E. A. Weiss, N. Maheswaranathan, and S. Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics, 2015. 1

[27] C. Team. Controlnetmediapipeface. https://github.com/ControlNet/ControlNetMediaPipeFace, 2023. 2

[28] O. team. Opencv: Open source computer vision library. https://docs.opencv.org/4.x/d1/dee/tutorial_introduction_to_openvino.html, 2023. Accessed: 2023-06-06. 2

[29] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. 4

[30] X. Xu, J. Guo, Z. Wang, G. Huang, I. Essa, and H. Shi. Prompt-free diffusion: Taking ”text” out of text-to-image diffusion models, 2023. 2

[31] L. Zhang, A. Rao, and M. Agrawala. Adding conditional control to text-to-image diffusion models, 2023. 1, 2, 4

[32] Z. Zhang. Utkfaces. https://susanqq.github.io/UTKFace/, 2017. Accessed: 2023-06-06. 4

[33] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks, 2020. 2