Not All Steps are Created Equal:
Selective Diffusion Distillation for Image Manipulation
Supplemental Material

Luozhou Wang$^{1,4*}$ Shuai Yang$^{1,4*}$ Shu Liu$^2$ Ying-cong Chen$^{1,3,4†}$

$^1$ The Hong Kong University of Science and Technology (Guangzhou). $^2$ SmartMore.
$^3$ The Hong Kong University of Science and Technology. $^4$ HKUST (Guangzhou) - SmartMore Joint Lab.

A. Implementation Detail

The image resolution for the human face dataset is set to $1024 \times 1024$. The image resolution for the cat face and car dataset is set to $512 \times 512$. The hyperparameters like learning rate, weight decay, loss weight, noise type, iterations, and mapper levels are adjusted individually for each experiment based on our experimental heuristics. More details can be found in our code. It is attached with the supplementary file. All experiments are conducted on an NVIDIA RTX3090, with 24GB memory. The architecture of the image manipulator is the same as the latent mapper of [5].

B. Diffusion model as a prior

As mentioned in Sec. 3.2, Diffusion models can also be used as off-the-shelf modules in some scenarios, where one model may be a prior for another conditional model. A typical example is a diffusion model $p(x)$ trained on MNIST digits and an off-the-shelf classifier $c(x, y)$ where $y$ is the class label. Then we can use the diffusion model to generate data of a specific class for the classifier. In theory, this means that we want to deduce $p(x \mid y)$ given $p(x)$ and $c(x, y)$. One solution is to introduce an approximate variational posterior $q(x)$ to approximate the posterior distribution $p(x \mid y)$, and minimize:

$$F = -E_{q(x)}[\log p(x) - \log q(x)] - E_{q(x)}[\log c(x, y)]$$  

When extending this formula to the scenario of diffusion model with latent variable $x_1, ..., x_T$, we can define this approximate variational posterior $q(x)$ as point estimate $q(x) = \delta(x - \eta)$ [2], and then minimize:

$$F = \sum_{t} E_{\epsilon \sim \mathcal{N}(0,1)}[\|\epsilon - \epsilon_0(x_t, t)\|^2_2] - E_{q(x)}[\log c(\eta, y)]$$  

$$x_t = \sqrt{\alpha_t} \eta + \sqrt{1 - \alpha_t} \epsilon.$$  

This equation optimizes $\eta$, which has the same dimensionality as data. We can regard this equation as directly sampling pixels using diffusion models to get a sample that satisfies the condition $y$.

Another situation is that $c(x, y)$ is a hard and non-differentiable conditional model, say a deterministic function $x = f(y)$. Then the gradient descent steps will be performed concerning $y$ on

$$\sum_{t} E_{\epsilon \sim \mathcal{N}(0,1)}[\|\epsilon - \epsilon_0(\sqrt{\alpha_t} f(y) + \sqrt{1 - \alpha_t} \epsilon, t)\|^2_2].$$  

For example, $f()$ can be a latent-variable model that takes latent $y$ as input and generates a sample $x$. Another example is that we can also use techniques of differentiable image parameterization. In [6], $y$ can be parameters of a 3D volume, and $f$ is a volumetric renderer. We can regard this equation as sampling $y$ instead of directly sampling images using diffusion models and inputting $y$ to this conditional model. We will get a sample from the diffusion model.

Optimizing simultaneously for all $t$ makes it difficult to guide the sample toward a mode. Thus existing methods either anneal $t$ from high to low values [2], or random select $t$ [6]. So the actual optimization process is slightly changed to

$$E_{\epsilon, t}[\|\epsilon - \epsilon_0(\sqrt{\alpha_t} f(y) + \sqrt{1 - \alpha_t} \epsilon, t)\|^2_2].$$  

C. Fine-grained control of image manipulation

Another benefit of our approach is that the control capability of the image manipulator can be used to control the semantics more precisely. We use StyleGAN [3] as the final generator of images, and the hierarchical nature of StyleGAN allows us to decompose more complex manipulations into different levels of manipulation. For example, we can adjust the manipulation effect of the image in three levels: coarse, medium, and fine. For better control, we used the most controlled StyleSpace [7] during the experiments. For more details, please watch the supplementary video.

---

*Equal contribution
†Corresponding author
D. More visual samples

In this section, we provide more visual samples (Fig. 1, Fig. 2, Fig. 3 and Fig. 4) from multiple domains. For different domains, the StyleGAN generator is pre-trained using different datasets [1, 4, 8]. We conduct various manipulation for the human face domain, including attribute and identity translation. Both results demonstrate the effectiveness of our methods.

E. Video

We summarize the analysis of selection and distillation, the fine-grained control of our method, and more visual samples in a supplementary video.

References

<table>
<thead>
<tr>
<th>Original image</th>
<th>“white hair”</th>
<th>“black hair”</th>
<th>“bald”</th>
<th>“bangs hair”</th>
<th>“happy”</th>
<th>“sad”</th>
</tr>
</thead>
</table>

Figure 1. Additional results for SDD face manipulation, in the aspect of hair color, hairstyle, and facial expression
Figure 2. Additional results for SDD face manipulation, in the aspect of attributes addition and celebrities conversion
<table>
<thead>
<tr>
<th>Original image</th>
<th>“British shorthair”</th>
<th>“black”</th>
<th>“Ragdoll”</th>
<th>“orange”</th>
<th>“big eyes”</th>
</tr>
</thead>
</table>

Figure 3. Additional results for SDD cat face manipulation
Figure 4. Additional results for SDD car manipulation