A Denoising Diffusion Implicit Model with non-zero mean latent space

The forward process of a diffusion process with non-zero mean latent distribution $\mathbf{y}_T \sim \mathcal{N}(f(\mathbf{x}), \mathbf{I}))$ has a closed form representation:

$$q(\mathbf{y}_t|\mathbf{y}_0, f) = \mathcal{N}(\mathbf{y}_t; \sqrt{\bar{\alpha}_t}\mathbf{y}_0 + (1 - \sqrt{\bar{\alpha}_t})f(\mathbf{x}), (1 - \bar{\alpha}_t)\mathbf{I}, \tag{A.1}$$

which could be reprameterized as:

$$\mathbf{y}_t = \sqrt{\bar{\alpha}_t}\mathbf{y}_0 + (1 - \sqrt{\bar{\alpha}_t})f(\mathbf{x}) + \sqrt{1 - \bar{\alpha}_t}\boldsymbol{\epsilon} \tag{A.2}$$

Then, similar to the DDIM, we define a Non-Markovian forward process with $\sigma_t \geq 0, t = 1 : T$.

$$q_\sigma(\mathbf{y}_{1:T}|\mathbf{y}_0, f) := q_\sigma(\mathbf{y}_T|\mathbf{y}_0, f) \prod_{t=2}^{T} q_\sigma(\mathbf{y}_{t-1}|\mathbf{y}_t, \mathbf{y}_0, f) \tag{A.3}$$

Where $q_\sigma(\mathbf{y}_T|\mathbf{y}_0, f) = \mathcal{N}(\mathbf{y}_T; \sqrt{\bar{\alpha}_T}\mathbf{y}_0 + (1 - \sqrt{\bar{\alpha}_T})f(\mathbf{x}), (1 - \bar{\alpha}_T)\mathbf{I})$ and for all $t > 1$.

$$q_\sigma(\mathbf{y}_{t-1}|\mathbf{y}_t, \mathbf{y}_0, f) = \mathcal{N}\left(\mathbf{y}_{t-1}; \sqrt{\bar{\alpha}_{t-1}}\mathbf{y}_0 + (1 - \sqrt{\bar{\alpha}_{t-1}})f(\mathbf{x}) + \sqrt{1 - \bar{\alpha}_{t-1} - \sigma_t^2} \cdot \tilde{\boldsymbol{\epsilon}}, \sigma_t^2\mathbf{I}\right), \tag{A.4}$$

$$\tilde{\boldsymbol{\epsilon}} = \frac{1}{\sqrt{1 - \bar{\alpha}_t}} \cdot \left(\mathbf{y}_t - \sqrt{\bar{\alpha}_t}\mathbf{y}_0 - (1 - \sqrt{\bar{\alpha}_t})f(\mathbf{x})\right)$$

We can prove that the sampling process defined by Eq. (A.3) and Eq. (A.4) has the same *marginal distribution* as the closed-form sampling process in Eq. (A.1) by the following Lemma:

Lemma A.1. *For $q_\sigma(\mathbf{y}_{1:T}|\mathbf{y}_0, f)$ defined in Eq. (A.3) and $q_\sigma(\mathbf{y}_{t-1}|\mathbf{y}_t, \mathbf{y}_0, f)$ defined in Eq. (A.4), we have:*

$$q_\sigma(\mathbf{y}_t|\mathbf{y}_0, f) = \mathcal{N}(\mathbf{y}_t; \sqrt{\bar{\alpha}_t}\mathbf{y}_0 + (1 - \sqrt{\bar{\alpha}_t})f(\mathbf{x}), (1 - \bar{\alpha}_t)\mathbf{I}) \tag{A.5}$$

Proof. Assume for any $t \leq T$, if Eq. (A.5) is true, the following is also true:

$$q_\sigma(\mathbf{y}_{t-1}|\mathbf{y}_0, f) = \mathcal{N}(\mathbf{y}_{t-1}; \sqrt{\bar{\alpha}_{t-1}}\mathbf{y}_0 + (1 - \sqrt{\bar{\alpha}_{t-1}})f(\mathbf{x}), (1 - \bar{\alpha}_{t-1})\mathbf{I}), \tag{A.6}$$

then we can prove the statement with an induction argument for t from T to 1, since the base case $(t = T)$ already holds.

First, we have that:

$$q_\sigma(\mathbf{y}_{t-1}|\mathbf{y}_0) := \int_{\mathbf{y}_t} q_\sigma(\mathbf{y}_t|\mathbf{y}_0, f) q_\sigma(\mathbf{y}_{t-1}|\mathbf{y}_t, \mathbf{y}_0, f) d\mathbf{y}_t, \tag{A.7}$$

and

$$q_\sigma(\mathbf{y}_t|\mathbf{y}_0, f) = \mathcal{N}(\mathbf{y}_t; \sqrt{\bar{\alpha}_t}\mathbf{y}_0 + (1 - \sqrt{\bar{\alpha}_t})f(\mathbf{x}), (1 - \bar{\alpha}_t)\mathbf{I}) \tag{A.8}$$

$$q_\sigma(\mathbf{y}_{t-1}|\mathbf{y}_t, \mathbf{y}_0, f) = \mathcal{N}\left(\mathbf{y}_{t-1}; \sqrt{\bar{\alpha}_{t-1}}\mathbf{y}_0 + (1 - \sqrt{\bar{\alpha}_{t-1}})f(\mathbf{x}) + \sqrt{1 - \bar{\alpha}_{t-1} - \sigma_t^2} \cdot \tilde{\boldsymbol{\epsilon}}, \sigma_t^2\mathbf{I}\right), \tag{A.9}$$

$$\tilde{\boldsymbol{\epsilon}} = \frac{1}{\sqrt{1 - \bar{\alpha}_t}} \cdot \left(\mathbf{y}_t - \sqrt{\bar{\alpha}_t}\mathbf{y}_0 - (1 - \sqrt{\bar{\alpha}_t})f(\mathbf{x})\right).$$

According to [63] Eq. (2.115), we have that $q_\sigma(\mathbf{y}_{t-1}|\mathbf{y}_t, \mathbf{y}_0, f)$ is Gaussian, with mean $\boldsymbol{\mu}_{t-1}$ and co-variance $\boldsymbol{\Sigma}_{t-1}$:

$$\begin{aligned}\boldsymbol{\mu}_{t-1} =& \sqrt{\bar{\alpha}_{t-1}}\mathbf{y}_0 + (1 - \sqrt{\bar{\alpha}_{t-1}})f(\mathbf{x}) \qquad (A.10)\\ &+ \sqrt{1 - \bar{\alpha}_{t-1} - \sigma_t^2}\left(\frac{\sqrt{\bar{\alpha}_t}\mathbf{y}_0 + (1 - \sqrt{\bar{\alpha}_t})f(\mathbf{x}) - \sqrt{\bar{\alpha}_t}\mathbf{y}_0 - (1 - \sqrt{\bar{\alpha}_t})f(\mathbf{x})}{\sqrt{1 - \bar{\alpha}_t}}\right)\\ =& \sqrt{\bar{\alpha}_{t-1}}\mathbf{y}_0 + (1 - \sqrt{\bar{\alpha}_{t-1}})f(\mathbf{x}). \qquad (A.11)\end{aligned}$$

$$\boldsymbol{\Sigma}_{t-1} = \sigma_t^2\mathbf{I} + \frac{1 - \bar{\alpha}_{t-1} - \sigma_t^2}{1 - \bar{\alpha}_t}(1 - \bar{\alpha}_t)\mathbf{I} = (1 - \bar{\alpha}_t)\mathbf{I}. \tag{A.12}$$

Therefore, Eq. (A.6) holds. Following the induction, the lemma is proved. □

In our implementation, we follow DDIM [32] setting $\sigma_t = 0$. The resulting model becomes an implicit probabilistic model [64], where the generation process become deterministic given $\mathbf{y}_T$.

B t-SNE visualization of the DDIM generation process

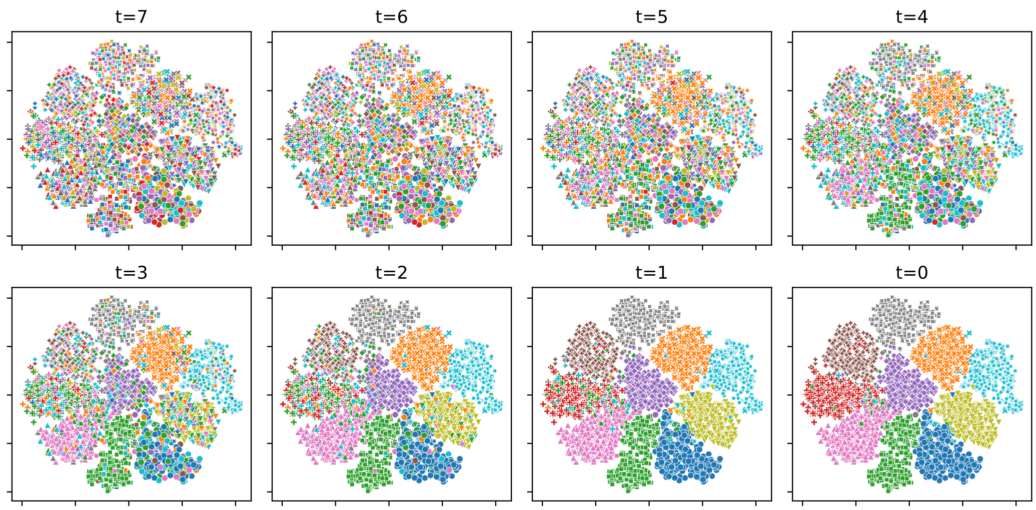

Figure B.1: The t-SNE visualization of the CLIP feature space during the reverse generation process of a conditional diffusion model using an 7-step DDIM on the CIFAR-10 dataset. The process begins at time $t = 7$, with the sampling of the latent representation of the label from the latent distribution $\mathcal{N}(f_q(\mathbf{x}), \mathbf{I})$. Through a series of multi-step reverse operations, the latent distribution is transformed into the conditional distribution of labels. The data points are color-coded according to the entry with the highest value in intermediate/final label vectors, and the ground truth class labels are represented by distinct markers.

C Experimental setup and details

C.1 Real-world dataset details

WebVision comprising 2.4 million images that were crawled using Google and Flickr search engines, with the ILSVRC12 taxonomy. Following prior studies, we trained our model on the initial 50 classes from the Google image subset of Webvision and tested it on the validation sets of both Webvision and ILSVRC12.

Food-101N consists of 310k food images collected from the internet with the Food-101 [65] taxonomy, and has an estimated label noise level of 20%, making it an ideal dataset to evaluate the robustness of our method under real-world noisy labels. We assessed the classification accuracy on the curated label set of Food-101, which contains around 25k images.

Clothing1M contains 1 million images of clothes obtained from shopping websites. Based on the keywords in the surrounding text, the images are automatically classified into 14 classes with $\sim$40% estimated noise level. The dataset includes a clean training set, validation set, and test set with manually refined labels, consisting of approximately 47.6k, 14.3k, and 10k pictures, respectively. We discarded the clean training set and only used the noisy label data for training.

C.2 Implementation details

To present the hyperparameter settings of our neural network, we first give a description of our neural network design. As shown in Figure C.1, the network consists of a frozen f_p encoder, a ResNet encoder, and a series of feed forward layers. Features encoded by the two encoders are combined with time embedding via hadamard product and passed through a series of feed-forward networks, batch normalization, and softplus activation to predict the noise term $\boldsymbol{\epsilon}_\theta$.

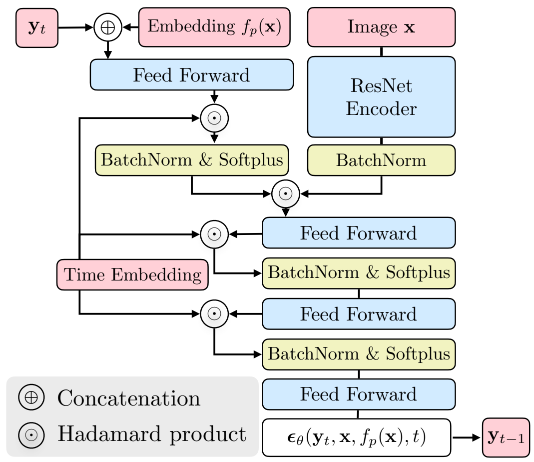

Figure C.1: The network architecture for conditional diffusion models. The input to the network consists of four elements: $\mathbf{y}_t$, $f_p(\mathbf{x})$, $\mathbf{x}$, and the time embedding for t, represented by pink blocks. The blue blocks in the figure represent the trainable network components.

In our experiments, we use ResNet34 for CIFAR10 and CIFAR100, and use ResNet50 for real-world datasets as the trainable encoder (blue ResNet block in Figure C.1). The dimensions of all feed-forward layers are set to 512 for CIFAR datasets and 1024 for real-world datasets, respectively. We train LRA-diffusion models for 200 epochs with Adam optimizer. The batch size is 256. We used a learning rate schedule that included a warmup phase followed by a half-cycle cosine decay. The initial learning rate is set to 0.001. Following [15], we applied data augmentation in the training, including resizing, random horizontal flip, and random cropping. To retrieve the nearest neighbors, we set k=10 based on our tests using a range of k values from 1 to 100 on the validation sets. The KNN accuracy remained relatively stable for k between 10 and 50, and then starts to decline due to reduced label consistency among neighbors. Based on these results, we infer that our LRA diffusion model is less sensitive to variations in k within this range. All experiments are conducted using four NVIDIA Titan V GPUs.

D Additional ablation study

D.1 Classifier feature conditioning for accuracy enhancement

To demonstrate how our method can enhance a trained classifier's performance by using its features as conditional information, we conduct an ablation study examining the impact of conditional diffusion and KNN on the trained classifier. Specifically, we train classifiers, denoted as $\eta(\mathbf{x})$, using the standard method at various noise levels. We then remove the classification head and utilize the remaining model f_η as the f_p encoders in our conditional diffusion models.

The experimental results shown in Figure D.1 indicate that these techniques can improve test accuracy. We observe that when the noise level is below 55%, the conditional diffusion model (green) achieves a $\sim 1\%$ improvement over the standard method. Moreover, when the LRA method is applied concurrently (purple), test accuracy can be further enhanced. This improvement occurs because learning from neighbors' labels reduces the noise level during training, as evidenced by the comparison between KNN results (blue) and clean label percentage (gray).

However, when the noise level exceeds 55%, the use of diffusion and LRA-diffusion methods does not seem advantageous. This limitation arises because the distribution of labels in the neighborhood becomes too corrupted for KNN to effectively improve the proportion of clean labels during training, as illustrated by the intersection of the blue and gray curves in the figure. We argue this does not diminish the practical value of our method because a dataset with more than 50% label noise is not meaningful in practice.

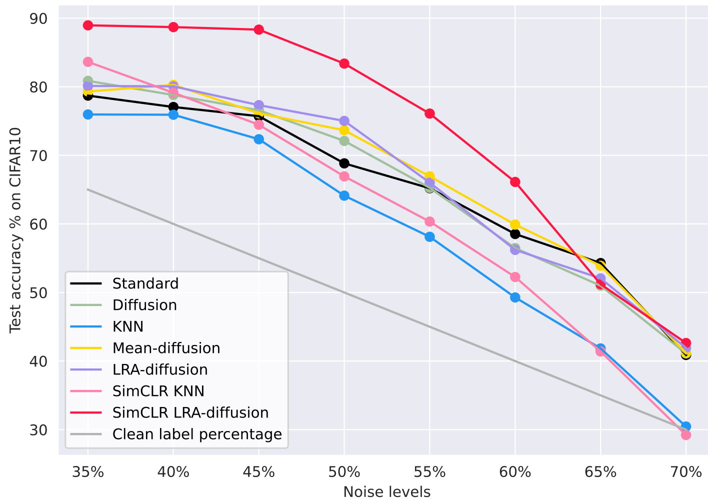

Figure D.1: Test accuracy of seven methods on the CIFAR-10 dataset with different levels of PMD label noise. Other than the already introduced method names, here *Diffusion* is a conditional diffusion model using the feature f_η. The clean label percentage is represented by the gray line.

D.2 Effects of different pseudo-label construction strategies

We also conduct comparative experiments using another method that utilizes neighbor labels: replacing the one-hot label vector with the mean vector of the neighbor's labels as the prediction target, which we call Mean-diffusion. We found that it can achieve higher accuracy when the noise level is higher than 55%. This may be due to the increase in the diversity of neighbor labels. The sampling-based LRA-diffusion will need to learn a more complex multi-modal distribution, but Mean-diffusion only needs to learn a point estimate. However, when the noise level is lower than 55%, we found that LRA-diffusion is slightly more accurate than Mean-diffusion. A possible explanation is that the distribution of $\mathbf{y}_0$ in LRA-diffusion contains only n one-hot labels. In contrast, $\mathbf{y}_0$ in Mean-diffusion is more diverse ($n^k/k!$ possible mean vectors for n classes and k neighbors). In conclusion, LRA-diffusion has higher performance with less noisy labels. On the other hand, Mean-diffusion has faster and more stable convergence and is more robust for high noise level. However, they tend to perform similarly when the noise level is too high or too low since neighbors' labels will become the same or too corrupted.

D.3 Robustness of pre-trained encoder conditioning

Finally, we use the SimCLR model as the encoder f_p in our conditional diffusion model (listed as SimCLR LRA-diffusion in Figure D.1), to showcase the effectiveness of our proposed LRA-diffusion method in utilizing prior knowledge from pre-trained image representations to enhance the test accuracy and robustness. The experimental results (red) show that its test accuracy significantly surpasses other settings until the noise level reaches 65%. Beyond this point, the labels in the neighborhood become too corrupted to provide additional supervision information.