Supplementary Information to Parametric Simulation of Electron Backscatter Diffraction Patterns through Generative Models

Zihao Ding[1] **and Marc De Graef**[1,*]

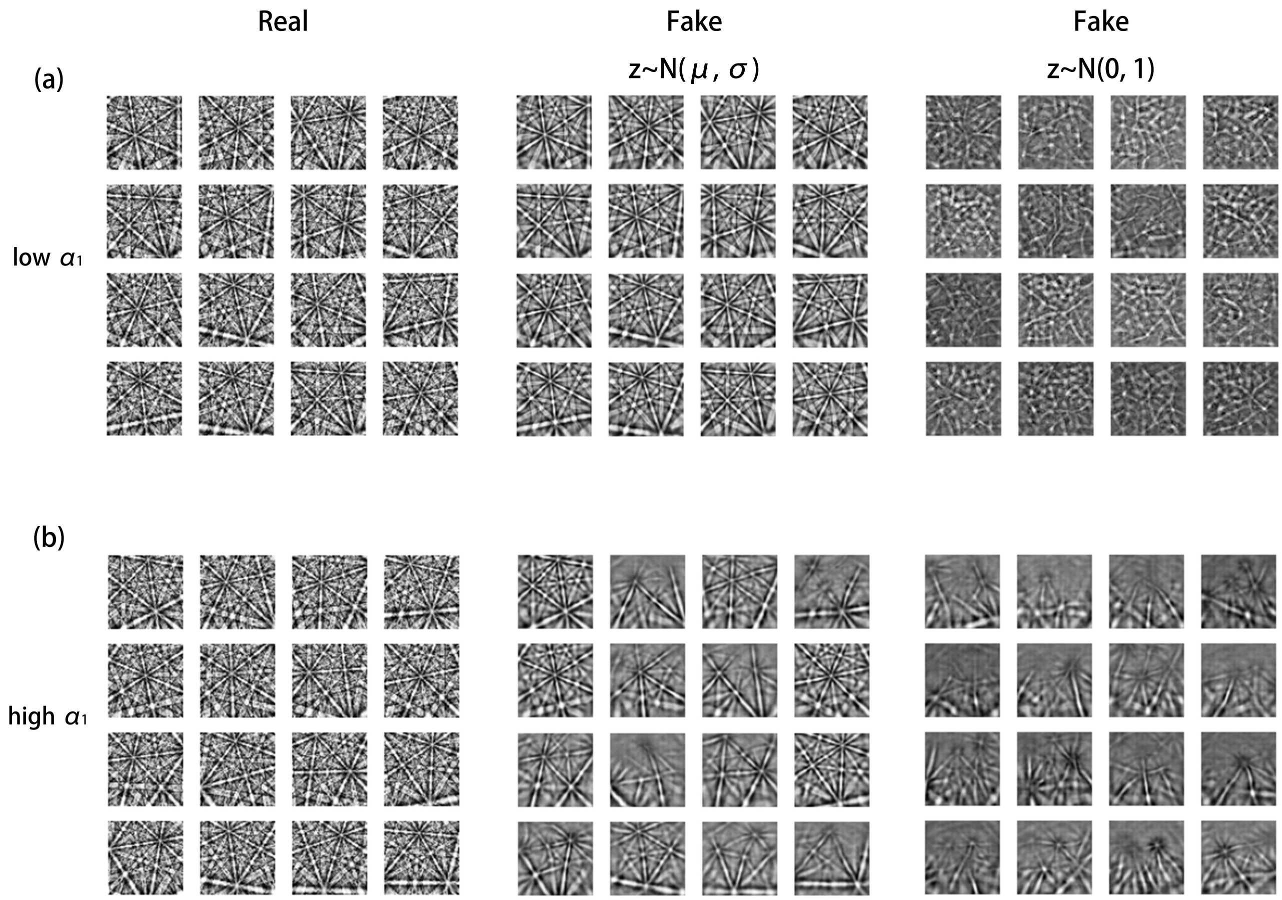
[1]Department of Materials Science and Engineering, Carnegie Mellon University, Pittsburgh, PA 15213, USA

[*]corresponding author(s): Marc De Graef (degraef@cmu.edu)

SUPPLEMENTARY NOTE 1. Adaptive Coefficient for KL Divergence

When optimization of the reconstruction loss is prioritized, the resulting generated patterns are shown in Supplementary Figure 1a. It can be seen that the band features, including both the contrast and location, are well reconstructed when the latent representations are drawn based on the normal distribution with parameters directly from the output of encoder ($z \sim N(\mu, \sigma)$). The only perceptible difference from patterns for training is that the generated patterns are slightly blurry on the edges, which is a known weakness of VAE[?]. But if the latent representation is forced to be drawn from a standard normal distribution ($z \sim N(0,1)$), only rambling band features can be observed in the generated patterns. Correspondingly, he Kullback–Leibler divergence (KL divergence) term was not minimized with a small coefficient in the overall loss expression, which indicates a poor pattern quality when the decoder/generator is deployed alone after the training and the latent representation $z \sim N(\mu, \sigma)$ is no longer available. Afterwards, the coefficient of the KL divergence in the loss was increased to prompt normal distribution in the latent space to a standard one, i.e. output of the encoder $(\mu, \sigma) \rightarrow (0, 1)$. It is observed that with a larger coefficient of L_1, the KL divergence decreases. The generated patterns with $\mathbf{z} \sim N(0,1)$ gradually approach those with $\mathbf{z} \sim N(\mu, \sigma)$, but the price is that the latter shows a quick drop in pattern quality. From Supplementary Figure 1b, the generated patterns with $\mathbf{z} \sim N(0,1)$ only maintain fuzzy band features, but the location is far from accurate.

According to patterns generated with the latent representation $\mathbf{z} \sim N(\mu, \sigma)$, the model size is proved

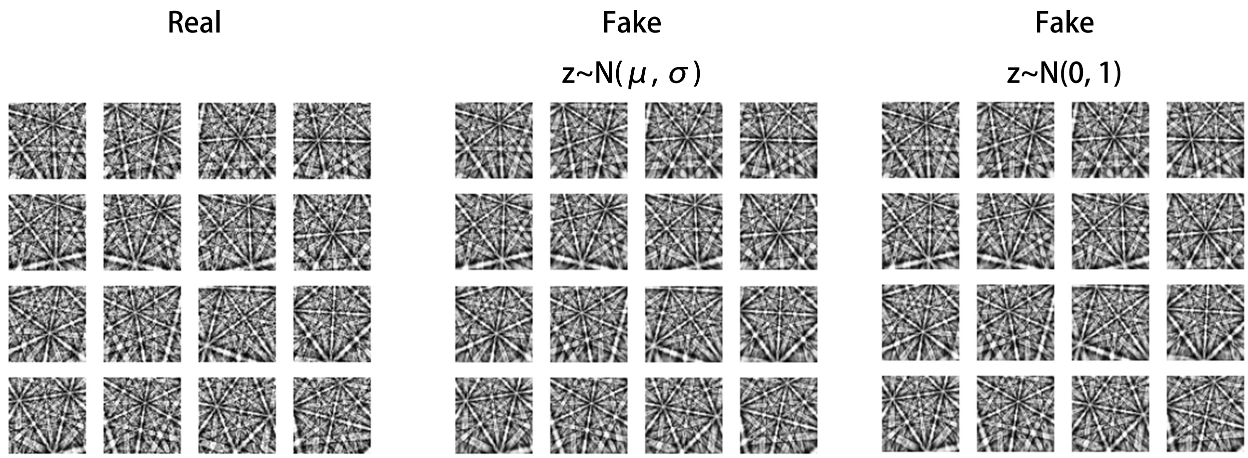


Supplementary Figure 1. Generated patterns with a latent representation size of 500: (a) reconstruction loss is prioritized to optimize, and (b) KL divergence is prioritized to optimize.

large enough. To handle the problem caused by a fixed coefficient which makes the model partial to either reconstruction loss or KL divergence and neglect the other one in the training, an adaptive coefficient is used to balance the reconstruction loss and the KL divergence. Here, a simple step function applied on α_1 is used:

$$\alpha_{1,new} = \alpha_1 * \begin{cases} 1.5 & L_1 \geq 15 \\ 0.1 & L_1 \geq 5 \\ 0.01 & else \end{cases} \tag{1}$$

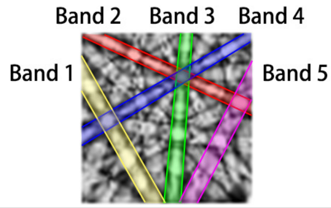
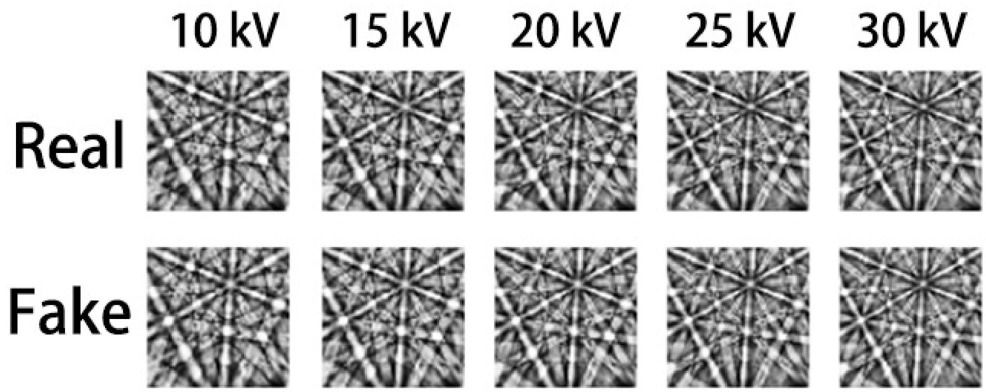
The final generated patterns are shown in Supplementary Figure 2.



Supplementary Figure 2. Generated patterns from a pure CVAE model with adaptive coefficient of KL divergence applied.

SUPPLEMENTARY NOTE 2. Width Change of Main Kikuchi Bands Caused by Accelerating Voltages

To analyze the effect of varying accelerating voltages, a random orientation $[56.697°, 58.856°, 325.550°]$ (in the form of zxz Euler angles $[\varphi_1, \Phi, \varphi_2]$) is picked up to fix the location of Kikuchi bands. Then EBSD patterns with 5 accelerating voltages are generated by both the forward model and the trained CVAE/GAN model. We measure the widths of 5 main Kikuchi bands and track their changes with increasing accelerating voltage. From the numbers in Supplementary Table 1, the Kikuchi bands do narrow when the incident beam has a higher energy, because the smaller Bragg angle changes the height/radius ratio of the Kossel cones, and, accordingly, the edges of the Kikuchi bands will move towards the band center. This phenomenon is well reproduced by the trained CVAE/GAN model, as the width of Kikuchi bands under different accelerating voltages are all very close to the ground truth. Due to the optimization from multiple aspects, including not only the pattern reconstruction error, but also the authenticity and the loss of each attribute, the model is able to fit the change caused by any of them.

Band					
Pattern					
Band 1 width, px					
AV, kV	10	15	20	25	30
Real	9.653 ± 0.693	7.756 ± 0.420	7.466 ± 0.354	6.75 ± 0.469	5.766 ± 0.410
Fake	9.561 ± 0.503	7.489 ± 0.494	7.155 ± 0.344	6.714 ± 0.368	5.723 ± 0.308
Band 2 width, px					
AV, kV	10	15	20	25	30
Real	6.253 ± 0.423	5.202 ± 0.237	4.840 ± 0.249	4.529 ± 0.245	4.42 ± 0.173
Fake	6.399 ± 0.373	5.181 ± 0.321	4.900 ± 0.404	4.570 ± 0.303	4.346 ± 0.337
Band 3 width, px					
AV, kV	10	15	20	25	30
Real	7.039 ± 0.408	6.329 ± 0.549	5.275 ± 0.277	4.602 ± 0.341	4.146 ± 0.261
Fake	6.915 ± 0.368	6.423 ± 0.362	5.242 ± 0.220	4.493 ± 0.239	4.107 ± 0.266
Band 4 width, px					
AV, kV	10	15	20	25	30
Real	6.238 ± 0.416	5.628 ± 0.451	4.242 ± 0.321	3.984 ± 0.292	3.648 ± 0.310
Fake	6.143 ± 0.327	5.46 ± 0.343	4.017 ± 0.267	3.876 ± 0.228	3.663 ± 0.333
Band 5 width, px					
AV, kV	10	15	20	25	30
Real	10.122 ± 0.535	9.391 ± 0.335	7.402 ± 0.431	6.821 ± 0.479	6.231 ± 0.311
Fake	10.269 ± 0.440	9.400 ± 0.469	7.423 ± 0.363	6.786 ± 0.361	6.122 ± 0.310

Supplementary Table 1. The width change of Kikuchi bands under different accelerating voltages.

SUPPLEMENTARY NOTE 3. EBSD-CVAE/GAN Training Hyperparameters

The training of EBSD-CVAE/GAN can be divided into two parts: 1) training of a CVAE model and discriminator/classifier of GAN separately, and 2) fine tuning of the whole model, where the generating part and the discriminating part are trained alternately. Hyperparameters used in each section are listed as follows:

Training of CVAE model

1. Batch size: 16;
2. Optimizer: Adam[?], with $\beta_1 = 0.5$ (the exponential decay rate for the 1st moment estimates), and $\beta_2 = 0.999$ (the exponential decay rate for the 2nd moment estimates);
3. Learning rate: 8×10^{-6};
4. Learning rate schedule: When validation loss does not improve in 2 epochs, the learning rate of next training epoch will be 50% of the current one. There is one epoch for cooldown after each decrease in learning rate;
5. Weight decay: 0 (The learning rate maintains in each epoch);
6. Loss function (using same definitions in the main body):

$$\begin{aligned} L &= \alpha_1 L_1 + \alpha_2 L_2 \\ &= 0.8 * L_1 + L_2 \end{aligned}$$

7. Stopping criteria: when learning rate is smaller than 5×10^{-7}.

Pre-training of Discriminator/classifier

1. Batch size: 64;
2. Optimizer: Adam[?], with $\beta_1 = 0.9$ (the exponential decay rate for the 1st moment estimates), and $\beta_2 = 0.999$ (the exponential decay rate for the 2nd moment estimates);

3. Learning rate: 5×10^{-4};

4. Weight decay: $\frac{0.25}{64}$ (To make the learning rate at the final of each epoch 80% of the learning rate at the beginning);

5. Loss function: crossentropy for discriminator, disorientation angle for orientation and MSE for accelerating voltage in the classifier;

6. Stopping criteria: 30 epochs.

Fine Tuning of CVAE/GAN model

1. Batch size: 8;

2. Training frequency: train discriminator/classifier once after every 9 iterations on CVAE part;

3. Optimizer: Adam[?], with $\beta_1 = 0.5$ (the exponential decay rate for the 1st moment estimates), and $\beta_2 = 0.999$ (the exponential decay rate for the 2nd moment estimates);

4. Learning rate: 4×10^{-6};

5. Learning rate schedule: When validation loss does not improve in 2 epochs, the learning rate of next training epoch will be 50% of the current one. There is one epoch for cooldown after each decrease in learning rate;

6. Weight decay: 0 (The learning rate maintains in each epoch);

7. Loss function (using same definitions in the main body):

   For iteration on CVAE:

$$\begin{aligned} L &= \alpha_1 L_1 + \alpha_2 L_2 + \alpha_{3,\mathbf{o}} L_{3,\mathbf{o}} + \alpha_{3,av} L_{3,av} + \alpha_{4,dec} L_{4,dec} \\ &= 0.8 * L_1 + L_2 + 0.1 * L_{3,\mathbf{o}} + 0.1 * L_{3,av} + 0.05 * L_{4,dec} \end{aligned}$$

For iteration on discriminator and classifier:

$$
\begin{aligned}
L &= \alpha_{3,\mathbf{o}} L_{3,\mathbf{o}} + \alpha_{3,av} L_{3,av} + \alpha_{4,dis} L_{4,dis} \\
&= L_{3,\mathbf{o}} + L_{3,av} + L_{4,dec}
\end{aligned}
$$

8. Stopping criteria: when learning rate is smaller than 5×10^{-7}.