Vector Quantized VAE for LArCV Images

Paul Lutkus
Department of Physics and Astronomy
Tufts University, Medford, MA

Neural Discrete Representation Learning: https://arxiv.org/abs/1711.00937
PixelCNN: https://arxiv.org/abs/1606.05328

Why Develop Generative Models?

- Fast simulation (NN’s are parallelizable)
- Learn useful representations via compression (200 Mb in 80Kb)
- Conditional generation (“Give me a picture of a…”)
- Anomaly detection ***

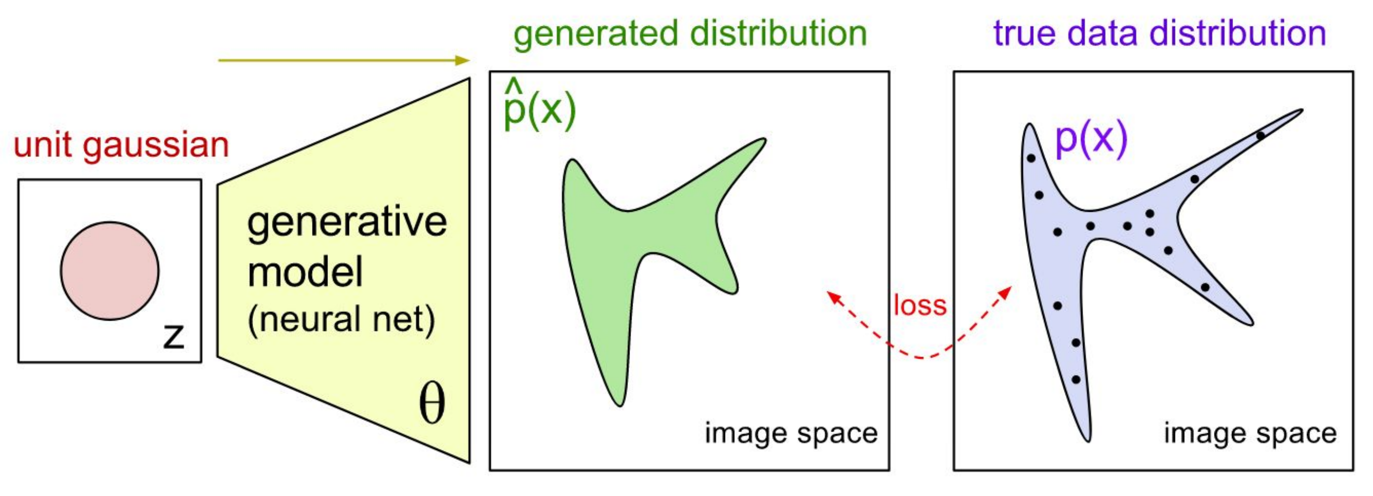


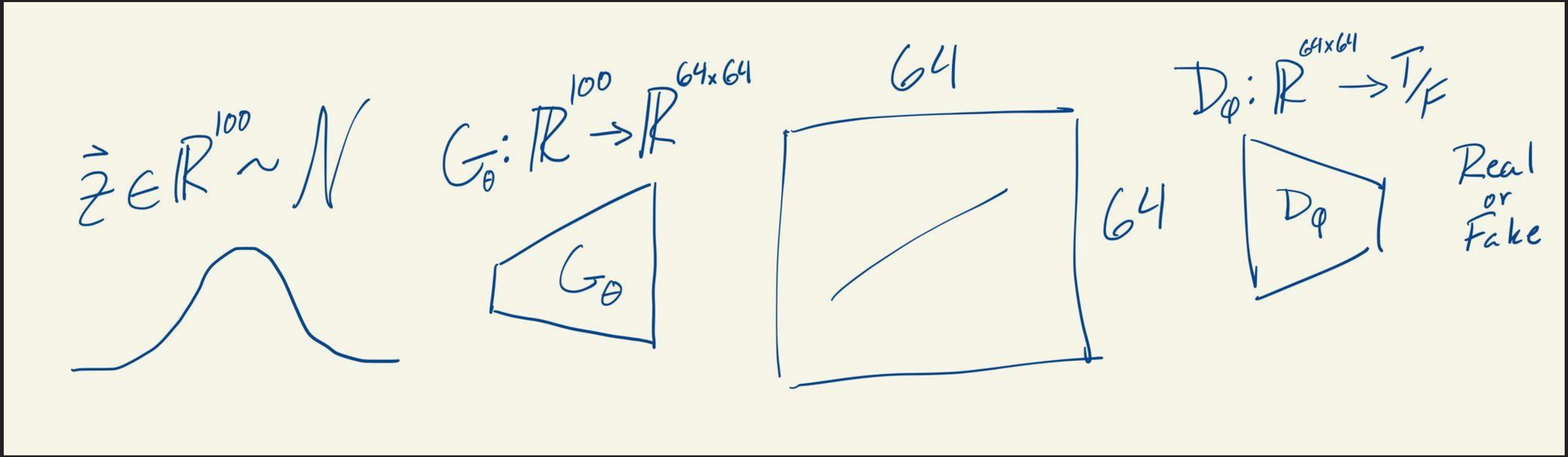
Figure:
[4] OpenAI

Generation Method: GAN

- Generative Adversarial Networks

Pros:
Discriminator training encourages realism

Cons:
Training is difficult



Generation Method: Autoregressive

$$p(\mathbf{x}) = \prod_{i=1}^{n^2} p(x_i | x_{i-1}, x_{i-2}, \ldots, x_1)$$

Pros:
Can model more complex dependencies

Cons:
Slow

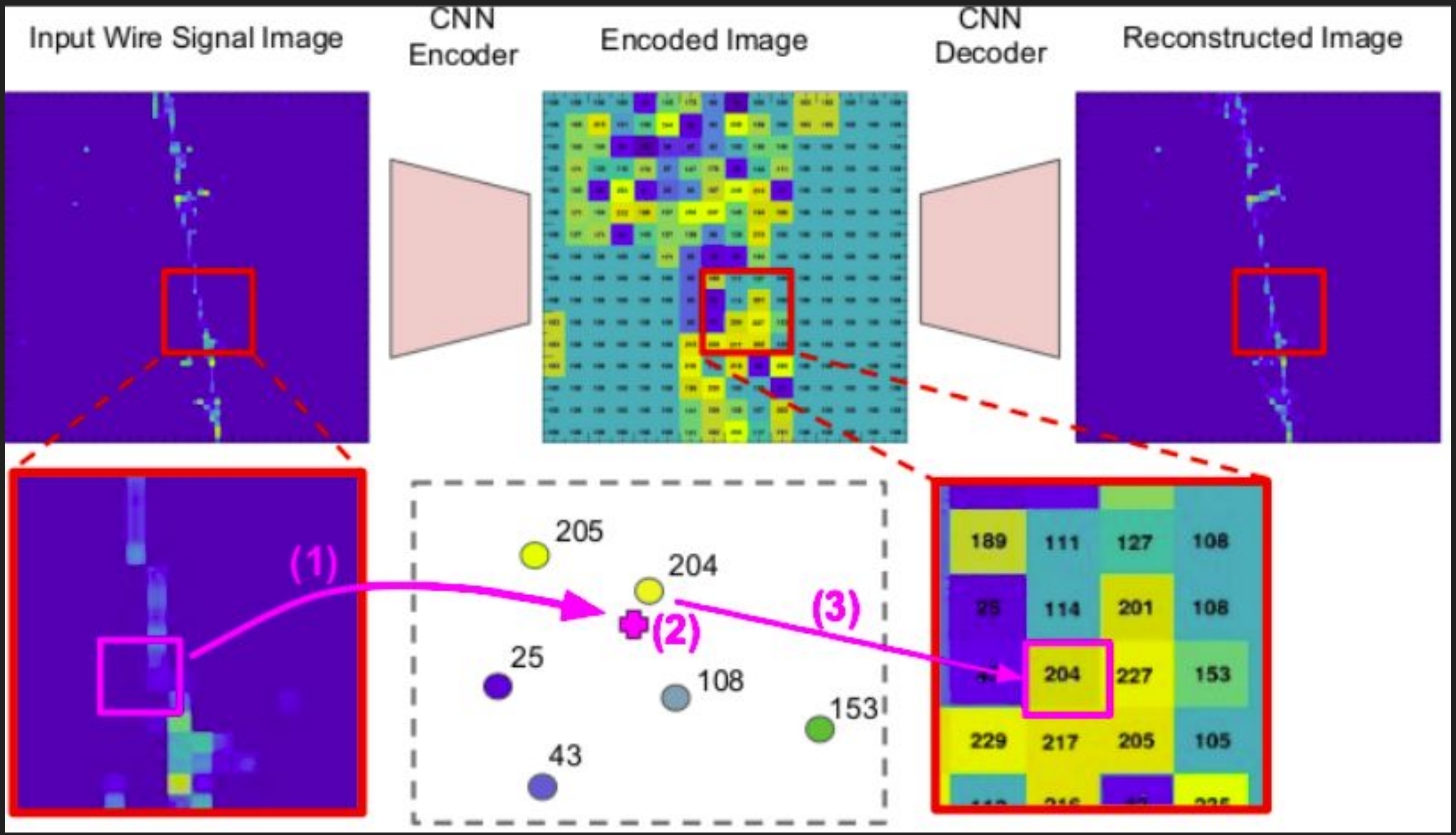
3-Pixel Image

$P(x_1)$	$P(x_2 \| x_1)P(x_1)$	$P(x_3 \| x_2)P(x_2 \| x_1)P(x_1)$

Generation Method: VQVAE

Pros:
- Feature selection
- Recurrence in feature space

Cons:
Still recurrent



Vector Quantization

$$V_i = \{\mathbf{x} : ||\mathbf{x} - \mathbf{y}_i|| \leq ||\mathbf{x} - \mathbf{y}_j||, \forall\ j \neq i\}$$

- Initialize code vectors randomly
- Cluster inputs around codes
- Average each cluster to get new codes

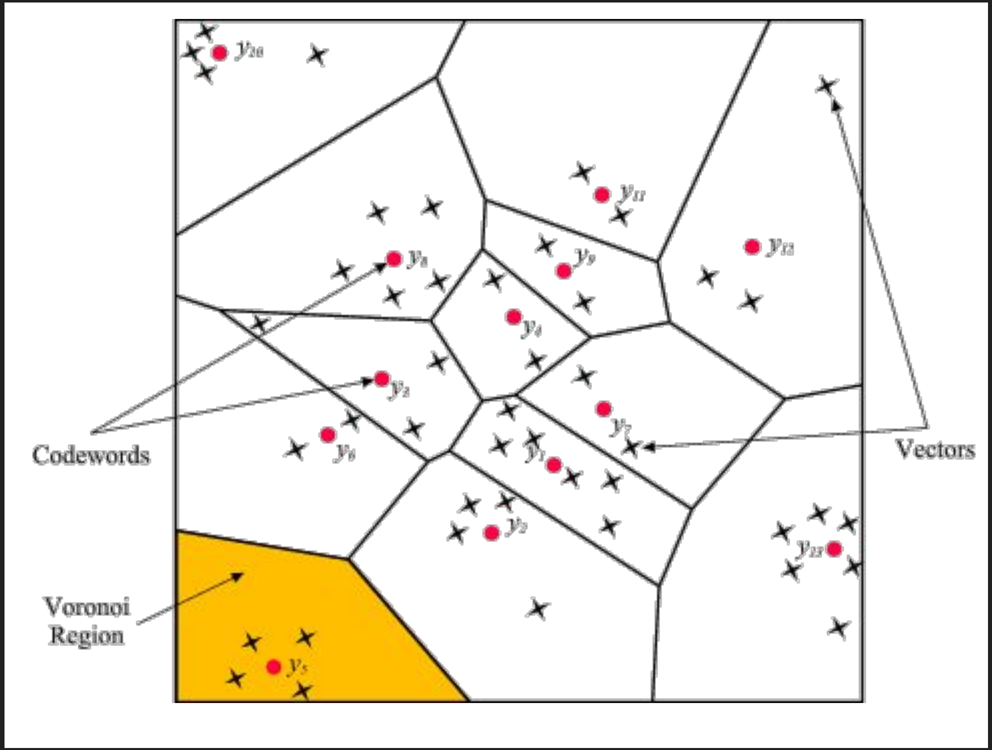


Figure: [3] Qasem M.

VQ for VQVAE

sg[] stops gradient flow

$$L_{\text{code}} = ||\text{sg}[z_e(\mathbf{x})] - z_q(\mathbf{x})||^2$$

$$L_{\text{commit}} = \beta||z_e(\mathbf{x}) - \text{sg}[z_q(\mathbf{x})]||^2$$

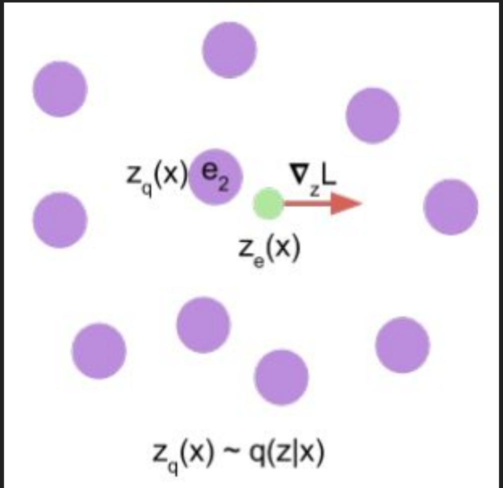


Figure: [1] Van der Oord et al. (arXiv:arXiv:1606.05328)

Pixelwise Quantization

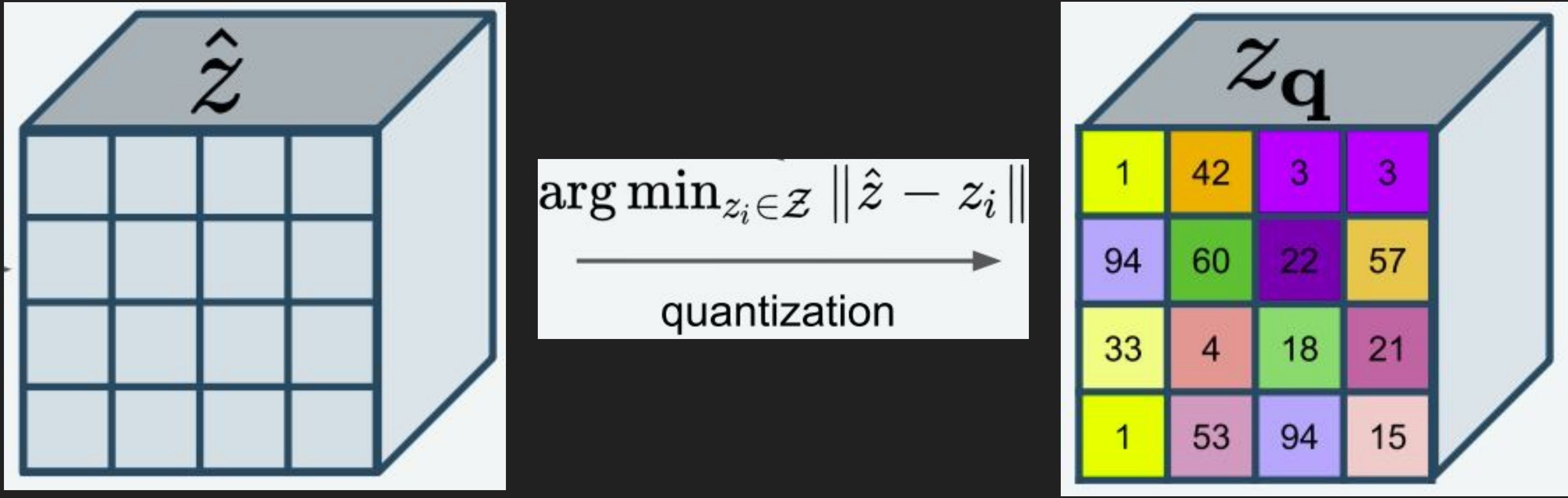
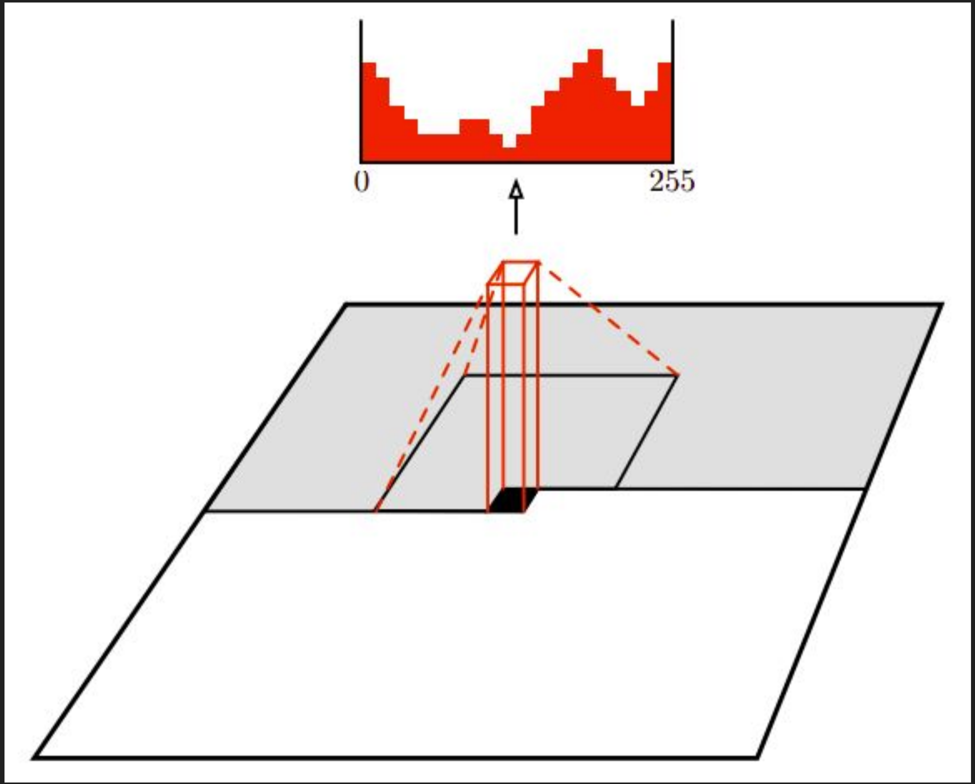


 Figure: Esser et al. (arXiv:2012.09841)

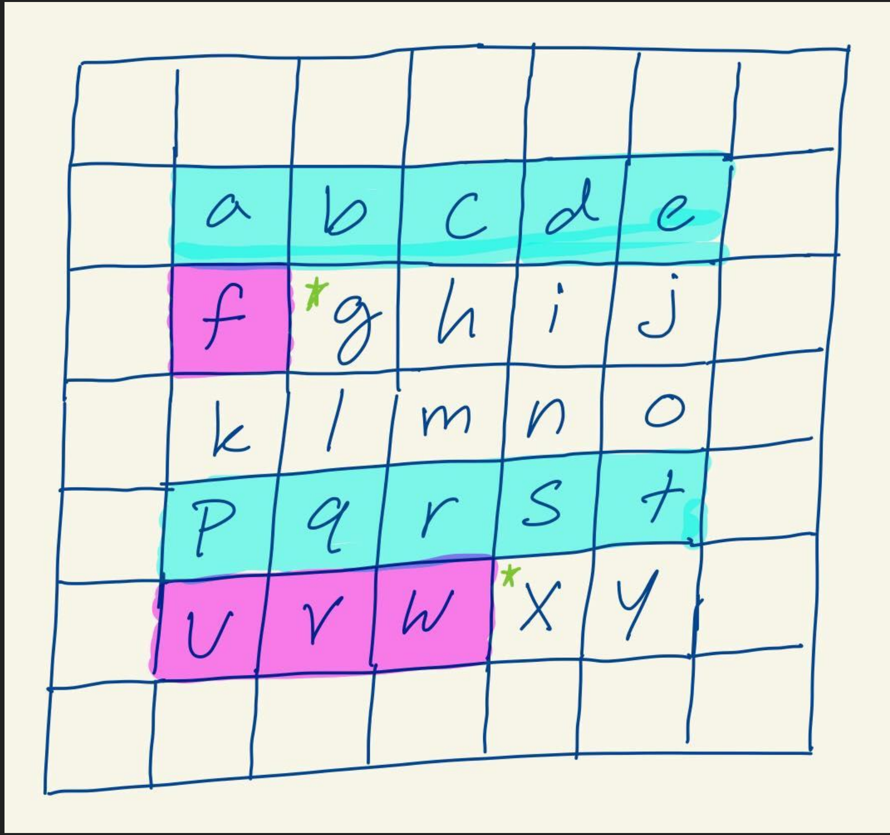
PixelCNN

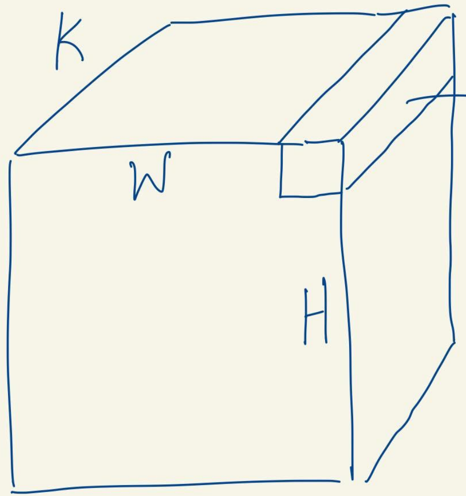


$$p(\mathbf{x}) = \prod_{i=1}^{n^2} p(x_i | x_{i-1}, x_{i-2}, \ldots, x_1)$$

Figure: [1] Van der Oord et al.
(arXiv:arXiv:1606.05328)

Modeling the Distribution





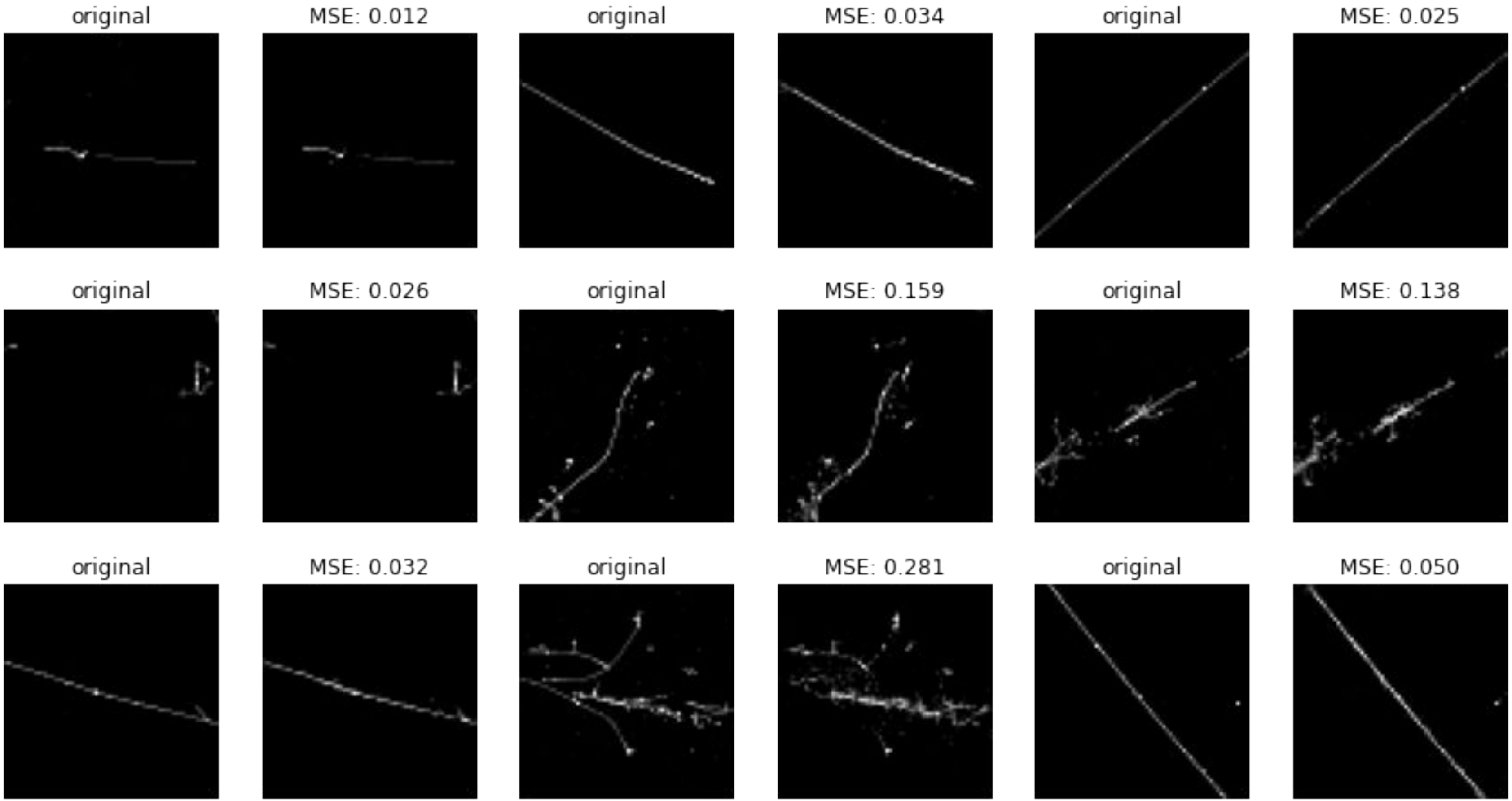
$$\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

$$i = 1, \ldots, K$$

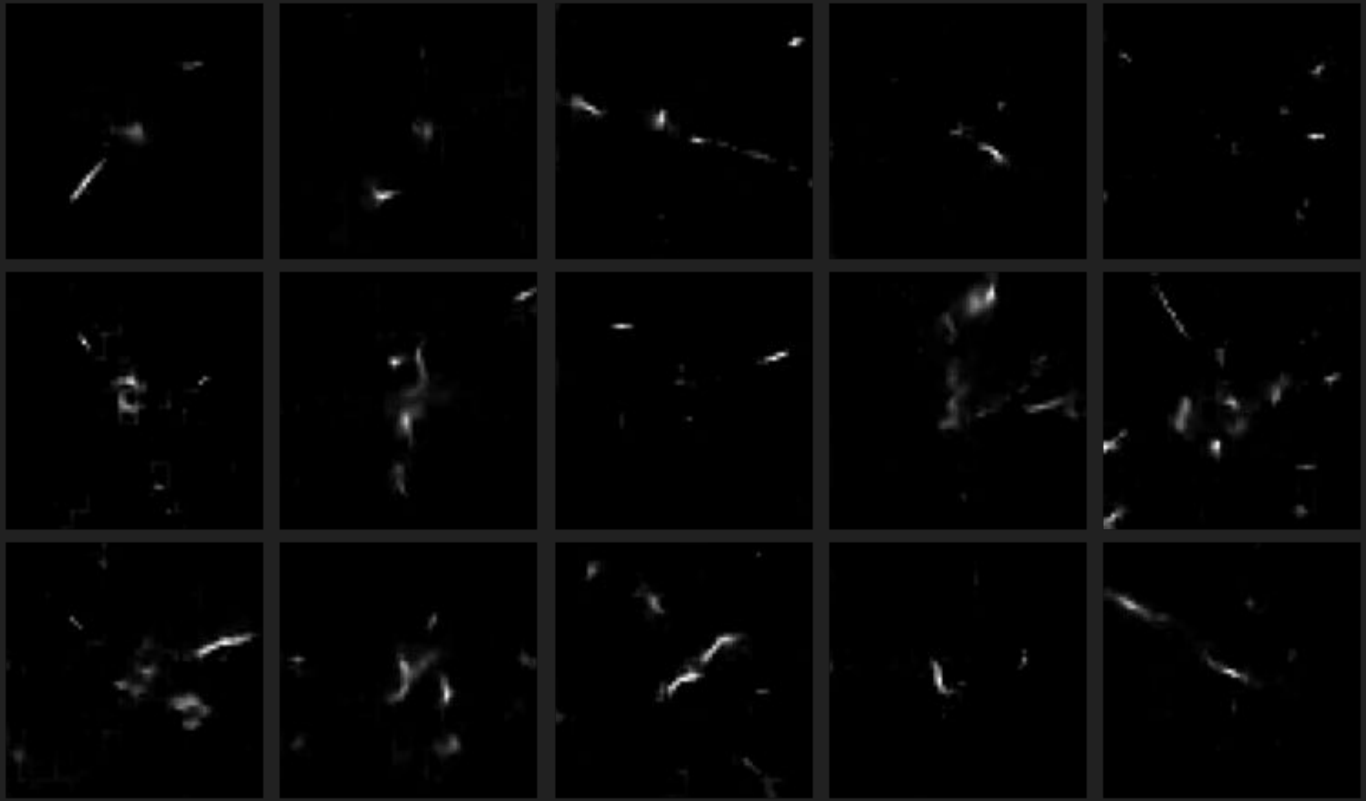
$$\vec{y} = \tanh(W_{k,f} * \vec{x}) \odot \sigma(W_{k,g} * \vec{x})$$

$*$: convolution, $\odot$: element-wise product

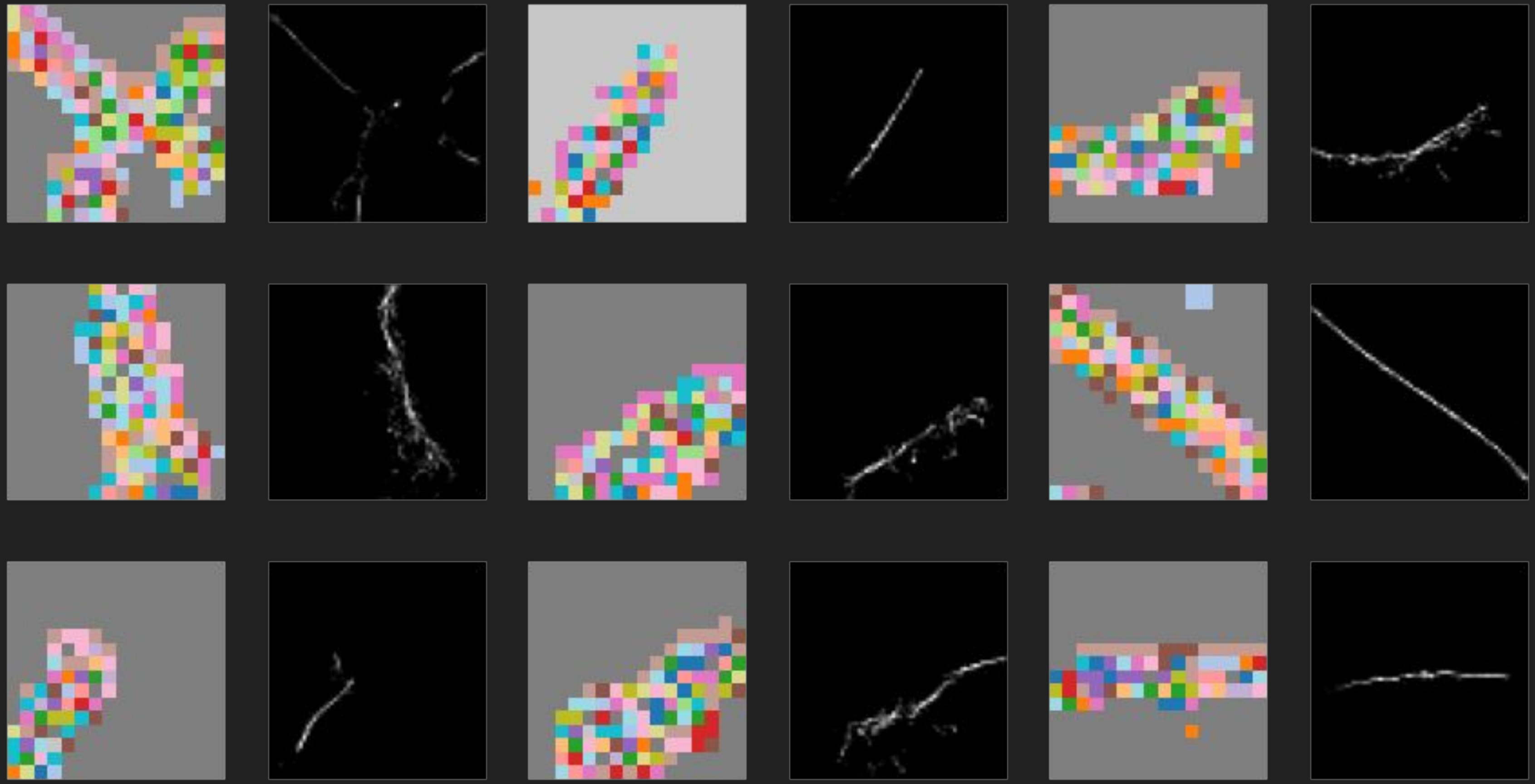
VQ-VAE Reconstructions | Avg. Test Loss: 0.071



VAE Samples

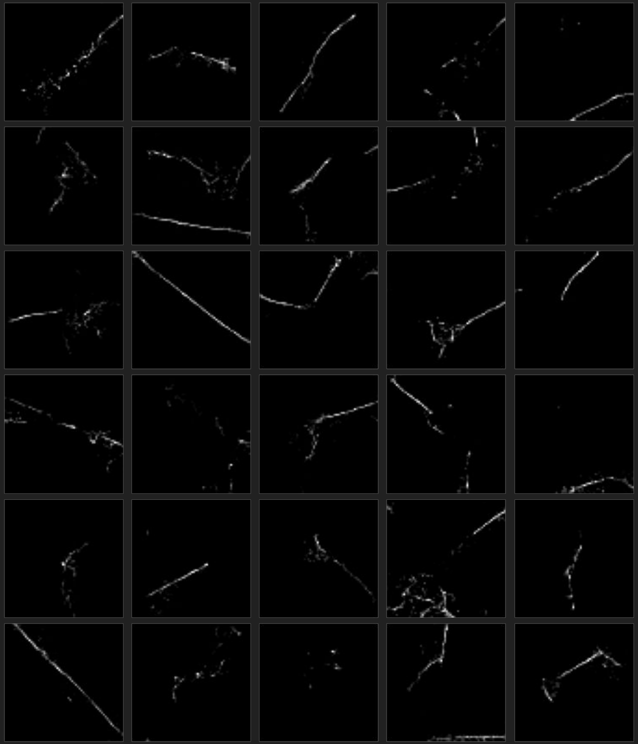


VQVAE Samples

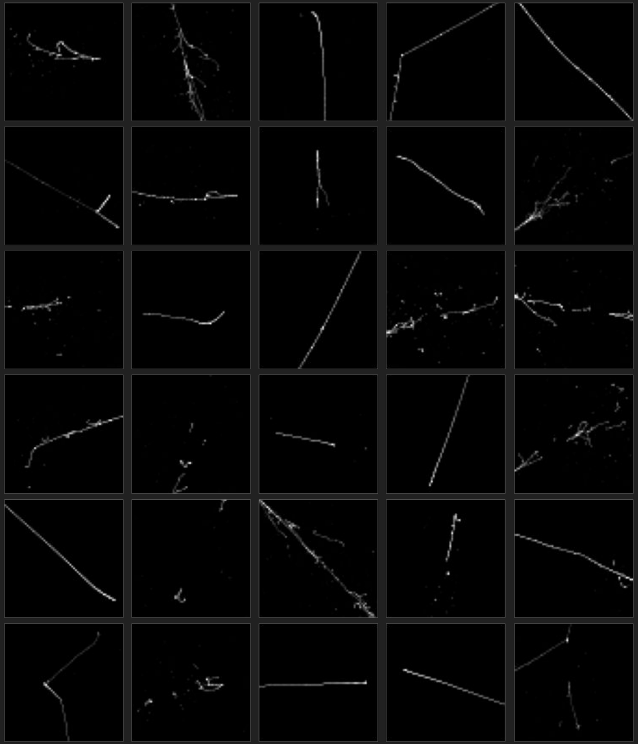


Generated vs Training

Generated Images



Training Images



Future Work

- Evaluation metrics
- Conditional generation
- VQVAE-2 and hierarchical representations
- PixelCNN -> Transformer
- Speed
- Discriminator

References

1. Conditional Image Generation with PixelCNN Decoders, Van der Oord et al. (arXiv:1606.05328)
2. Neural Discrete Representation Learning, Van der Oord et al. (arXiv:1711.00937)
3. Qasem, Mohamed. Vector Quantization. http://www.mqasem.net/vectorquantization/vq.html.
4. “Generative Models.” OpenAI, 16 June 2016, https://openai.com/blog/generative-models/.

Hyperparameters

VQ-VAE Hyperparameters (Decoder Filters are Reversed)

Batch Size	Epochs	Learning Rate (Adam)	Layer Filters (Enc.)	K	D	β
512	50	3e-4	[16, 32, 8]	256	8	1

PixelCNN Hyperparameters

Batch Size	Epochs	Learning Rate (Adam)	Gated Blocks	Block Width
512	50	1e-3	6	256

Blindspot