UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

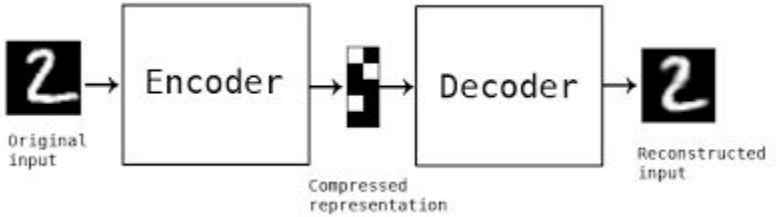
> Reinventa el mundo <

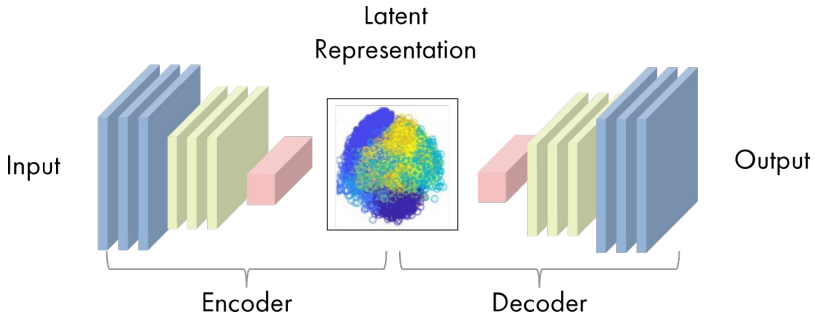
Deep Learning

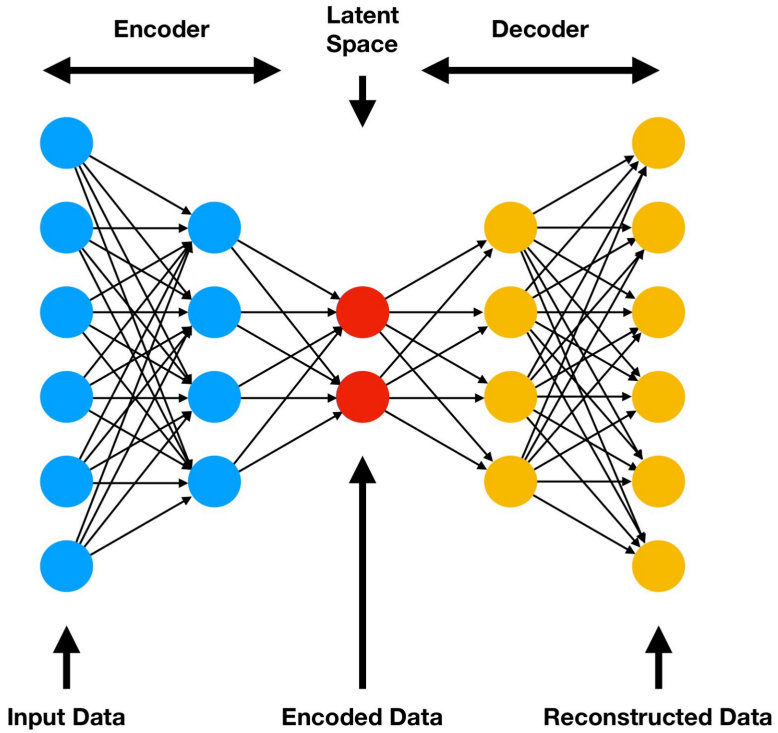
Week 6 : Variational AutoEncoders + Compression + Perceptual Optimization

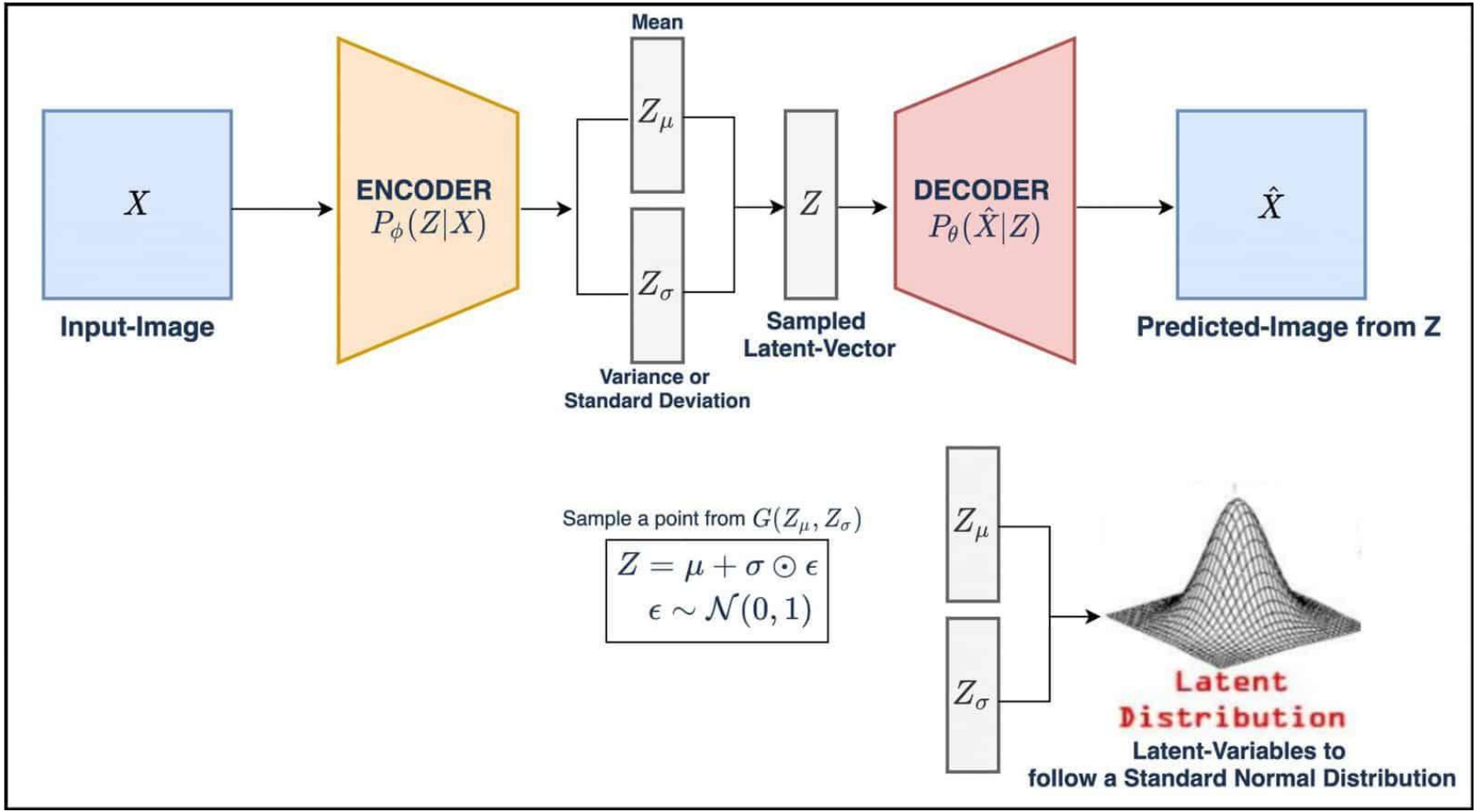
TRANSFORMATEC

Auto-Encoders Review








Mean
X
Input-Image
ENCODER
$P_\phi(Z|X)$
Z_μ
Z_σ
Variance or
Standard Deviation
Z
Sampled
Latent-Vector
DECODER
$P_\theta(\hat{X}|Z)$
$\hat{X}$
Predicted-Image from Z
Sample a point from $G(Z_\mu, Z_\sigma)$
$Z = \mu + \sigma \odot \epsilon$
$\epsilon \sim \mathcal{N}(0,1)$
Z_μ
Z_σ
Latent
Distribution
Latent-Variables to
follow a Standard Normal Distribution

Auto-Encoding Variational Bayes

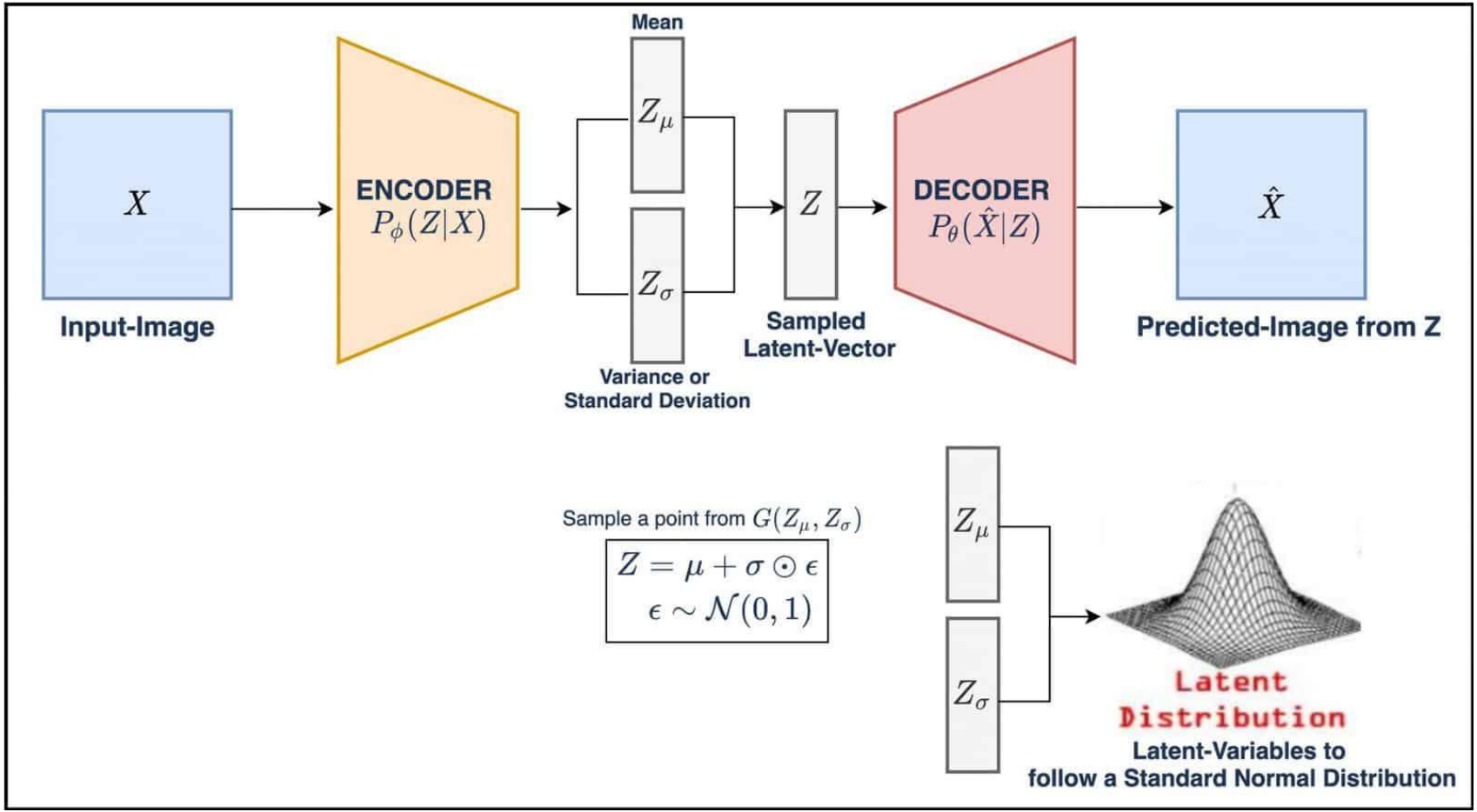
Diederik P. Kingma
Machine Learning Group
Universiteit van Amsterdam
dpkingma@gmail.com

Max Welling
Machine Learning Group
Universiteit van Amsterdam
welling.max@gmail.com

Abstract

Can we efficiently learn the parameters of directed probabilistic models, in the presence of continuous latent variables with intractable posterior distributions, and in case of large datasets? We introduce a novel learning and approximate inference method that works efficiently, under some mild conditions, even in the on-line and intractable case. The method involves optimization of a stochastic objective function that can be straightforwardly optimized w.r.t. all parameters, using standard gradient-based optimization methods.

The method does not require the typically expensive sampling loops per data-point required for Monte Carlo EM, and all parameter updates correspond to optimization of the variational lower bound of the marginal likelihood, unlike the wake-sleep algorithm. These theoretical advantages are reflected in experimental results.


Mean
X
Input-Image
ENCODER
$P_\phi(Z|X)$
Z_μ
Z_σ
Variance or
Standard Deviation
Z
Sampled
Latent-Vector
DECODER
$P_\theta(\hat{X}|Z)$
$\hat{X}$
Predicted-Image from Z
Sample a point from $G(Z_\mu, Z_\sigma)$
$Z = \mu + \sigma \odot \epsilon$
$\epsilon \sim \mathcal{N}(0,1)$
Z_μ
Z_σ
Latent
Distribution
Latent-Variables to
follow a Standard Normal Distribution

White-Board Derivation of Variational Auto-Encoder

But First : 1) What is KL-Divergence ?

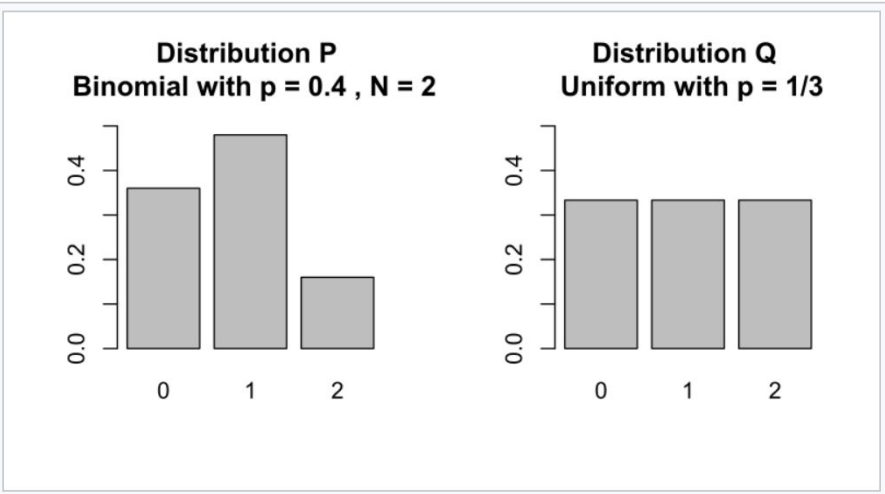
For two discrete pdf's p(x) and q(x) sampled over x we define the Kullback-Leibler Divergence from q(x) to p(x) as:

$$D_{\mathrm{KL}}(P \parallel Q) = \sum_{x \in \mathcal{X}} P(x) \log\left(\frac{P(x)}{Q(x)}\right),$$

which is equivalent to

$$D_{\mathrm{KL}}(P \parallel Q) = -\sum_{x \in \mathcal{X}} P(x) \log\left(\frac{Q(x)}{P(x)}\right).$$

KL - Divergence



Two distributions to illustrate relative entropy

Relative entropies $D_{\mathrm{KL}}(P \parallel Q)$ and $D_{\mathrm{KL}}(Q \parallel P)$ are calculated as follows. This example uses the natural log with base e, designated $\ln$ to get results in nats (see units of information):

$$\begin{aligned} D_{\mathrm{KL}}(P \parallel Q) &= \sum_{x \in \mathcal{X}} P(x) \ln\left(\frac{P(x)}{Q(x)}\right) \\ &= \frac{9}{25}\ln\left(\frac{9/25}{1/3}\right) + \frac{12}{25}\ln\left(\frac{12/25}{1/3}\right) + \frac{4}{25}\ln\left(\frac{4/25}{1/3}\right) \\ &= \frac{1}{25}\left(32\ln(2) + 55\ln(3) - 50\ln(5)\right) \approx 0.0852996, \\ D_{\mathrm{KL}}(Q \parallel P) &= \sum_{x \in \mathcal{X}} Q(x) \ln\left(\frac{Q(x)}{P(x)}\right) \\ &= \frac{1}{3}\ln\left(\frac{1/3}{9/25}\right) + \frac{1}{3}\ln\left(\frac{1/3}{12/25}\right) + \frac{1}{3}\ln\left(\frac{1/3}{4/25}\right) \\ &= \frac{1}{3}\left(-4\ln(2) - 6\ln(3) + 6\ln(5)\right) \approx 0.097455. \end{aligned}$$

x	0	1	2
Distribution $P(x)$	$\frac{9}{25}$	$\frac{12}{25}$	$\frac{4}{25}$
Distribution $Q(x)$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$

$$\mathrm{JSD}(P \parallel Q) = \frac{1}{2}D(P \parallel M) + \frac{1}{2}D(Q \parallel M),$$

where $M = \frac{1}{2}(P + Q)$ is a mixture distribution of P and Q.

Jensen Shannon Divergence

Note 1:

$$\mathrm{JSD}(P \parallel Q) = \frac{1}{2} D(P \parallel M) + \frac{1}{2} D(Q \parallel M),$$

where $M = \frac{1}{2}(P + Q)$ is a mixture distribution of P and Q.

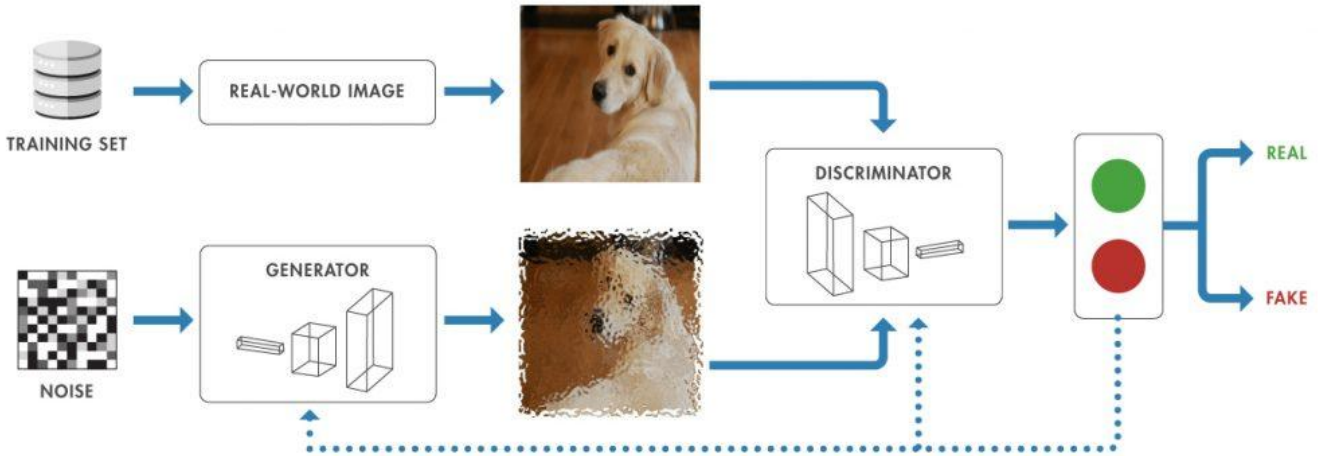
Note 2:

With base-e logarithm, which is commonly used in statistical thermodynamics, the upper bound is $\ln(2)$. In general, the bound in base b is $\log_b(2)$:

$$0 \leq \mathrm{JSD}(P \parallel Q) \leq \log_b(2).$$

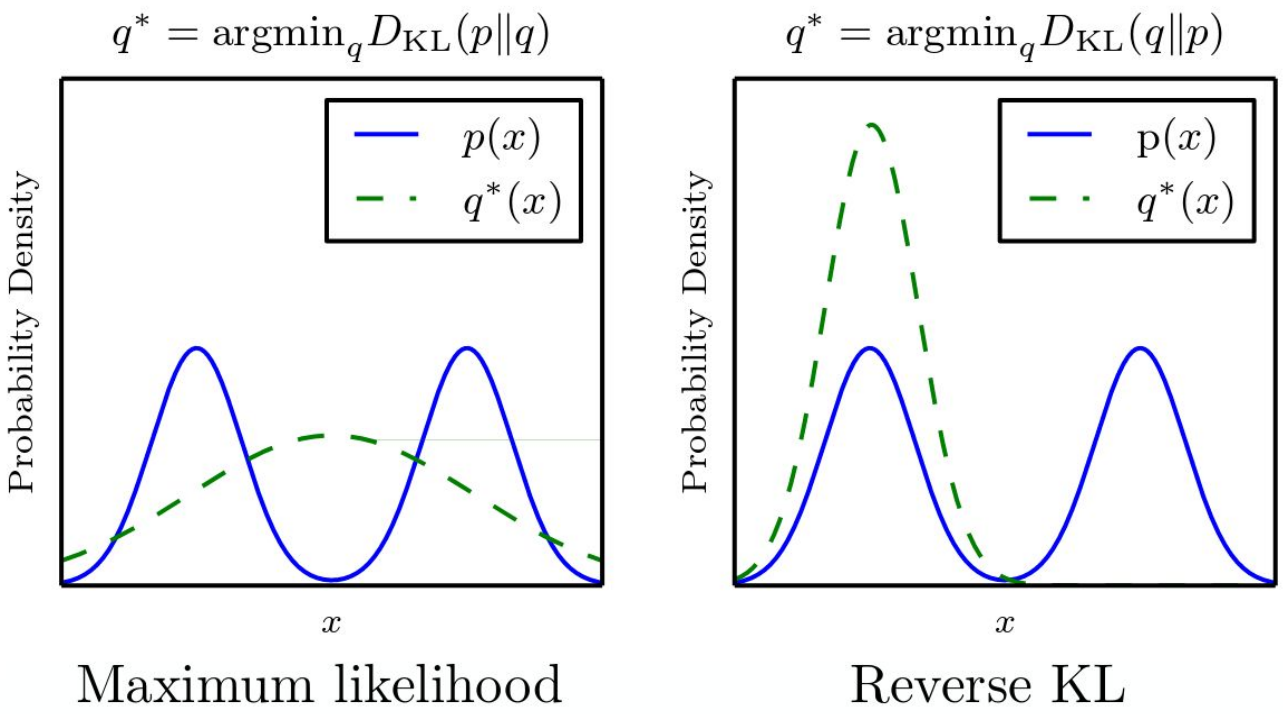
Note 3: JSD (P||Q) = JSD(Q||P)
Symmetrical Property!

Generative Adversarial Networks

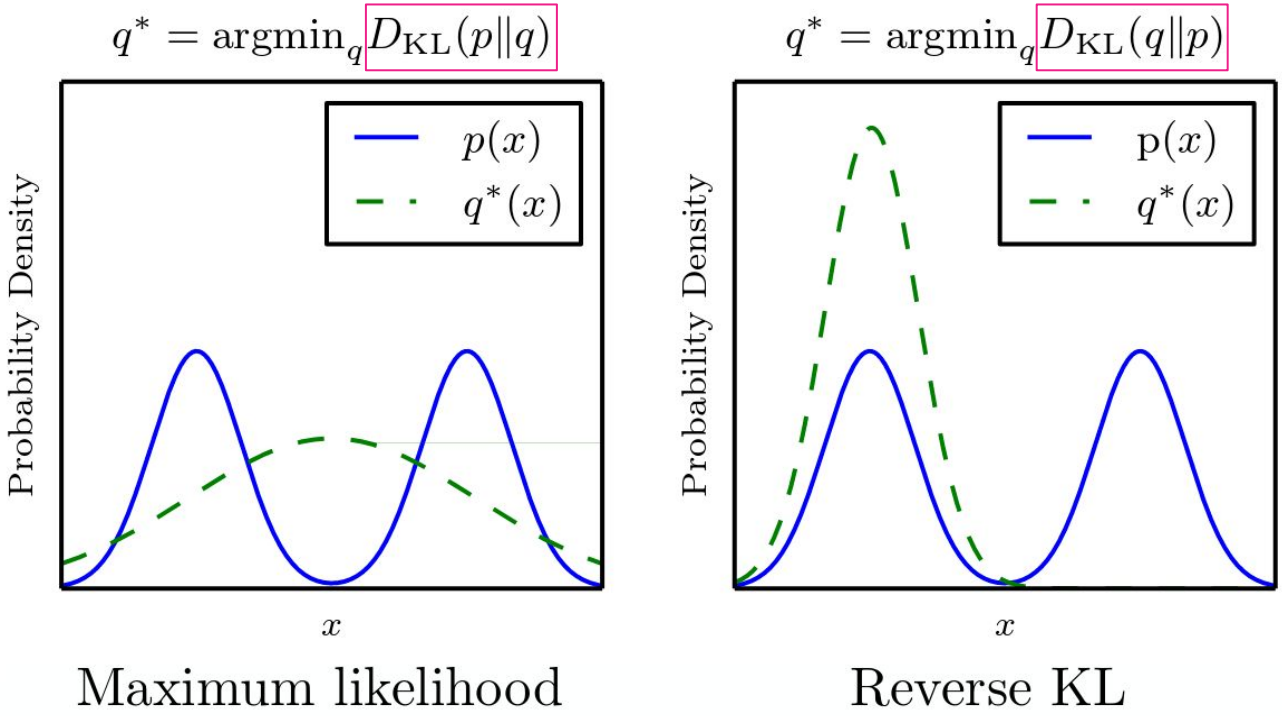


$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))].$$

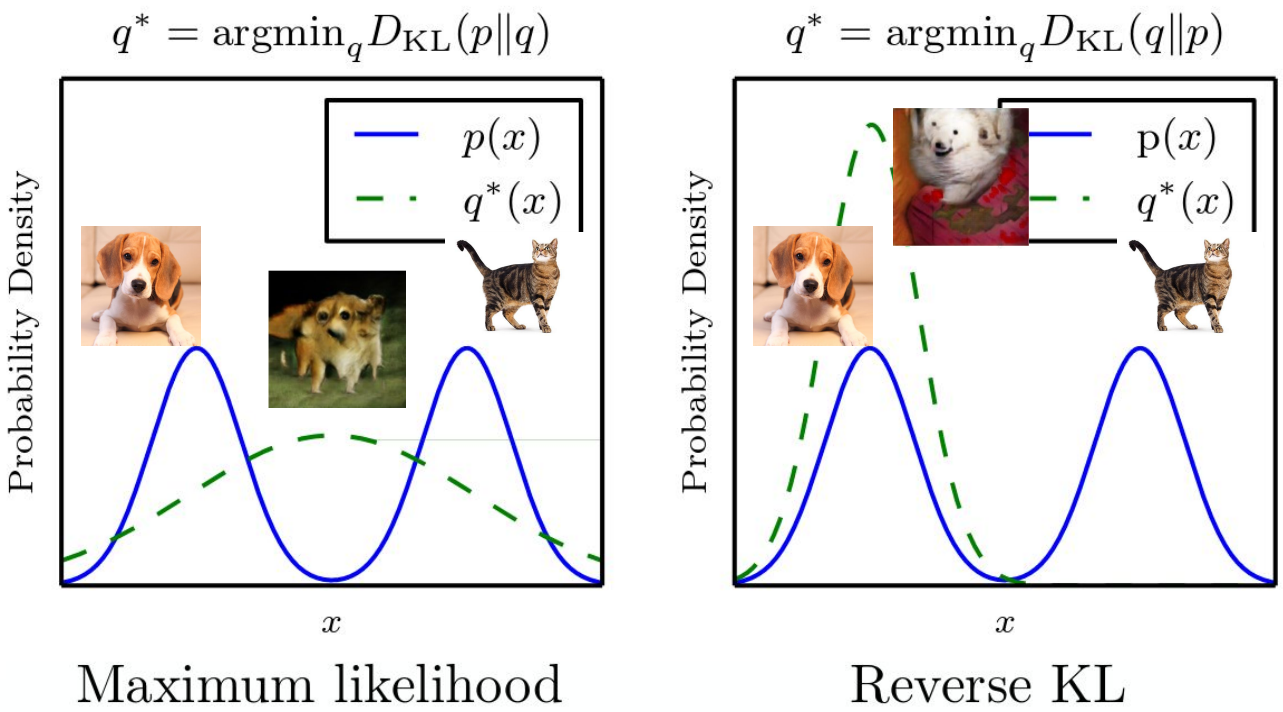
Consider *p(x)* being the real data distribution,
and *q*(x)* being the generated data distribution



Consider *p(x)* being the real data distribution,
and *q*(x)* being the generated data distribution

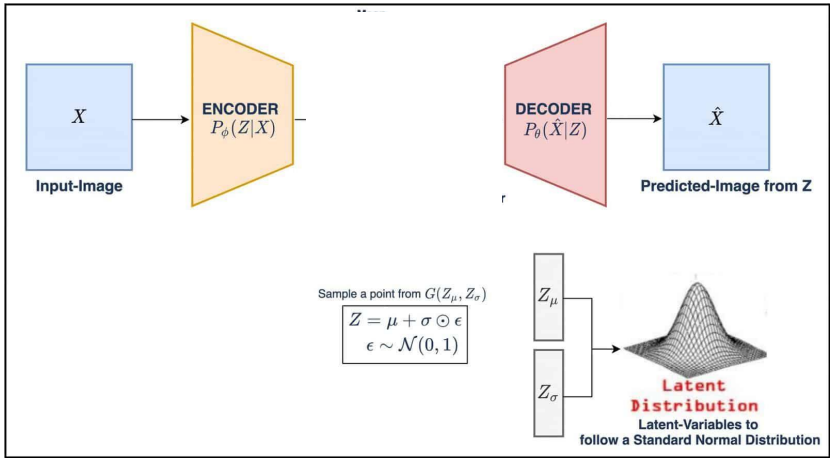


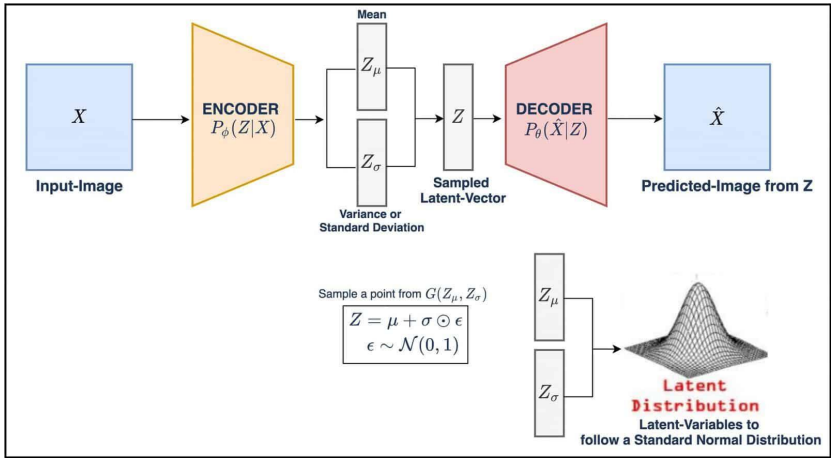
Consider *p(x)* being the real data distribution,
and *q*(x)* being the generated data distribution

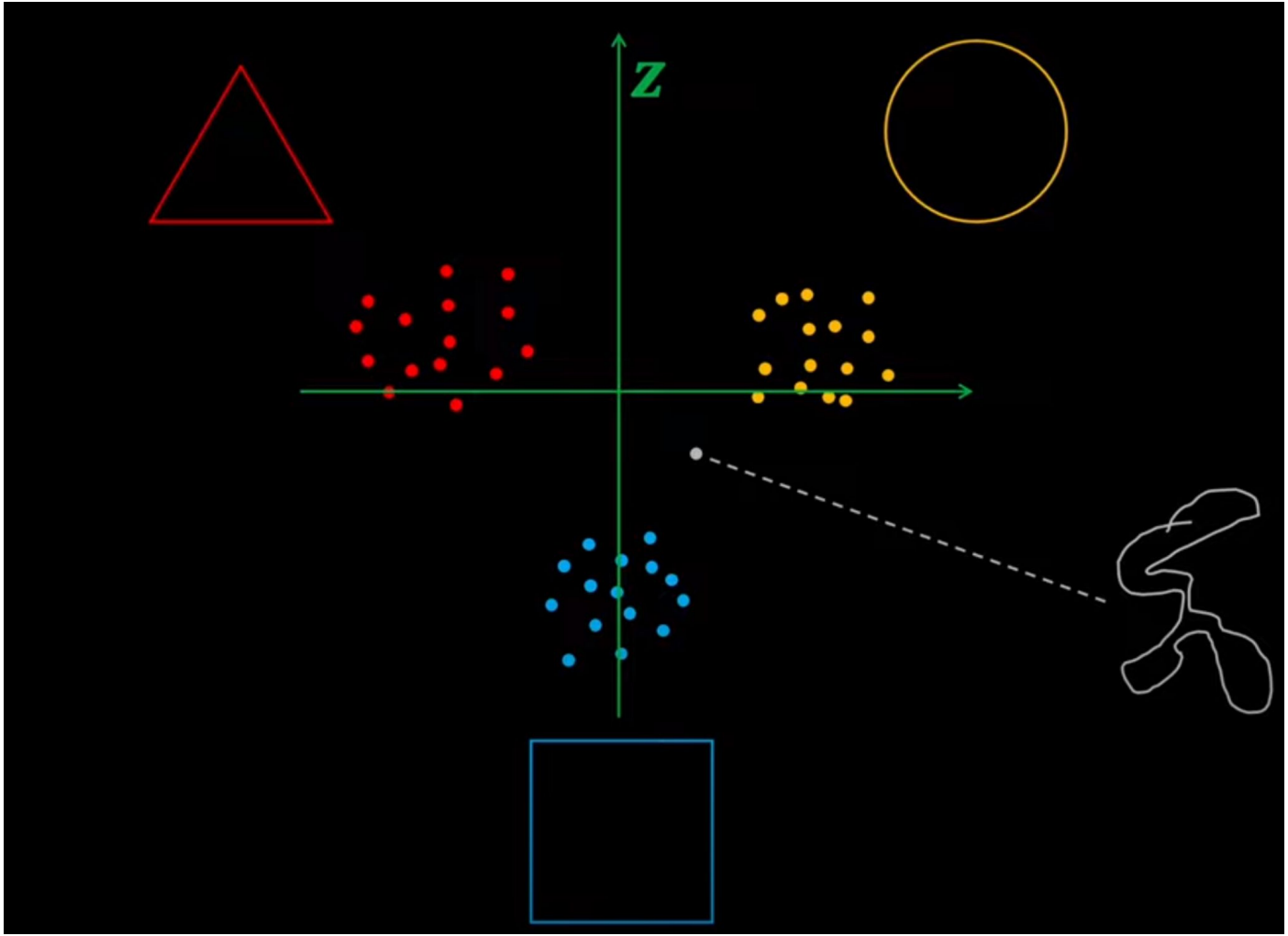


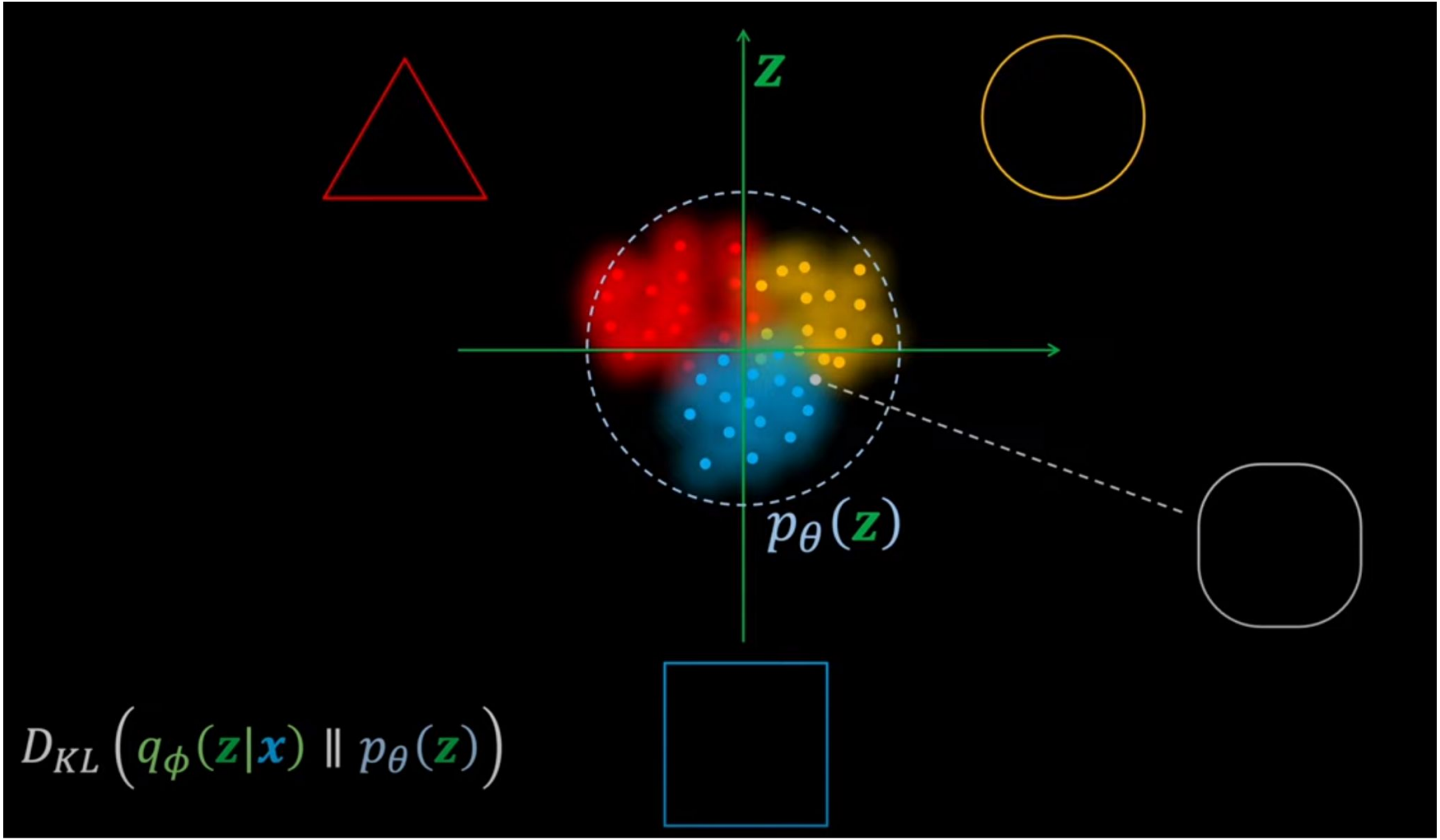
"Mode Dropping"

Differences + Similarities between AE's + VAE's

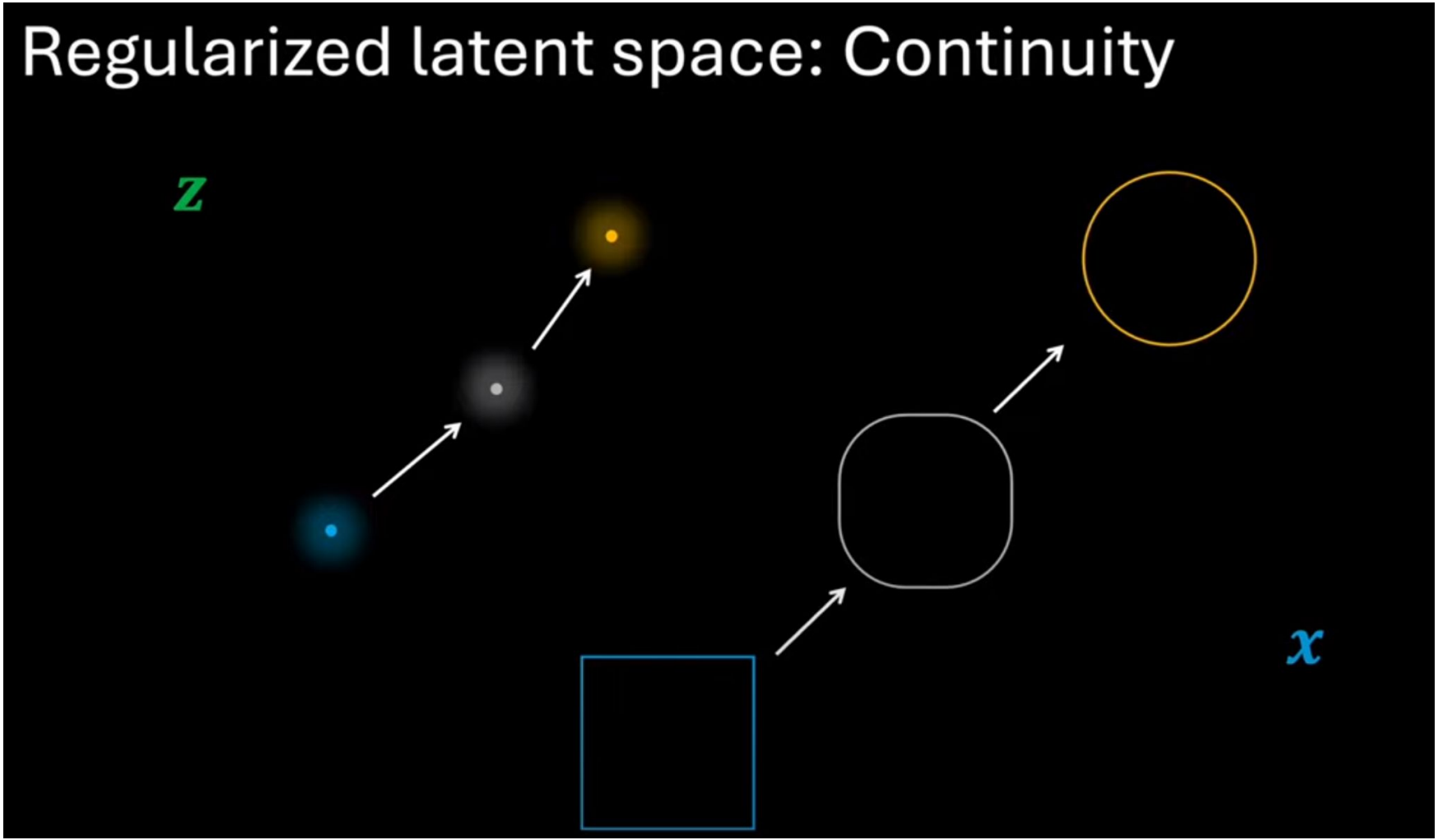





z


z
$p_\theta(z)$
$D_{KL}\left(q_\phi(z|x) \parallel p_\theta(z)\right)$

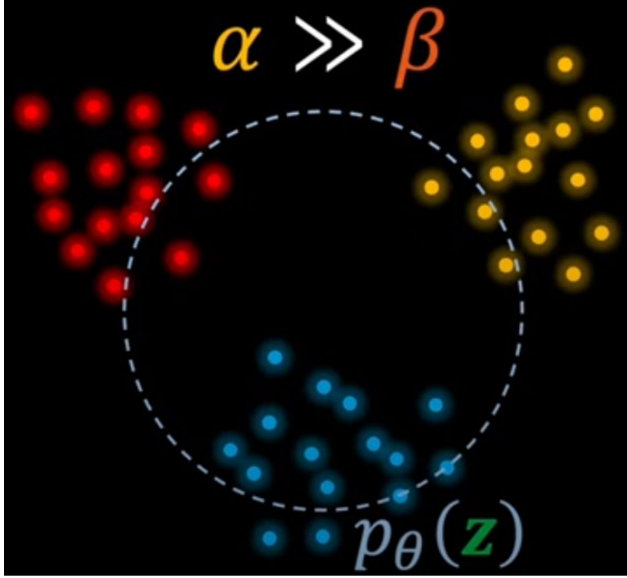
Regularized latent space: Continuity



Loss balance

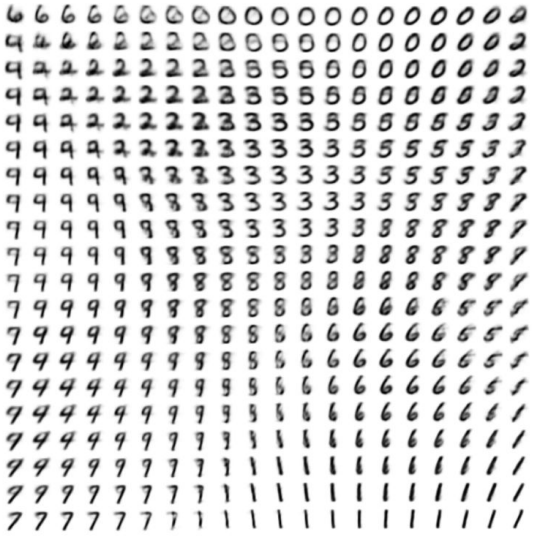
VAE loss $\mathcal{L} = \mathcal{L}_{\text{recons}} + \mathcal{L}_{\text{KL}}$

Weighted loss $\mathcal{L} = \alpha\mathcal{L}_{\text{recons}} + \beta\mathcal{L}_{\text{KL}}$



Variational Auto-Encoders

(a) Learned Frey Face manifold



(b) Learned MNIST manifold

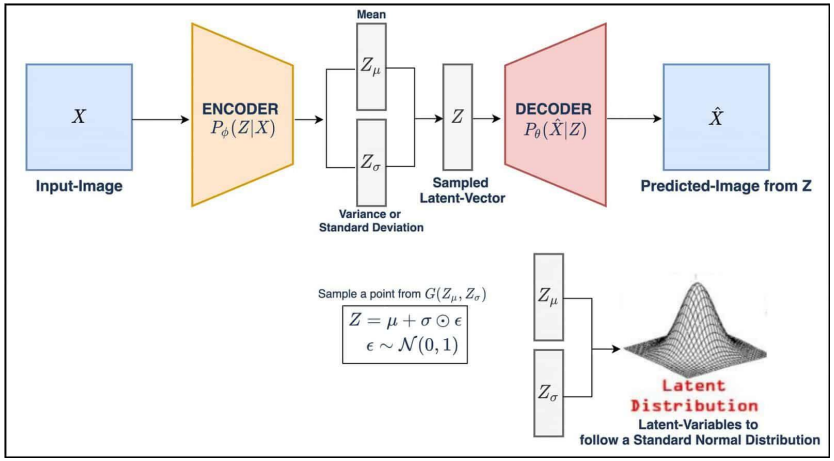
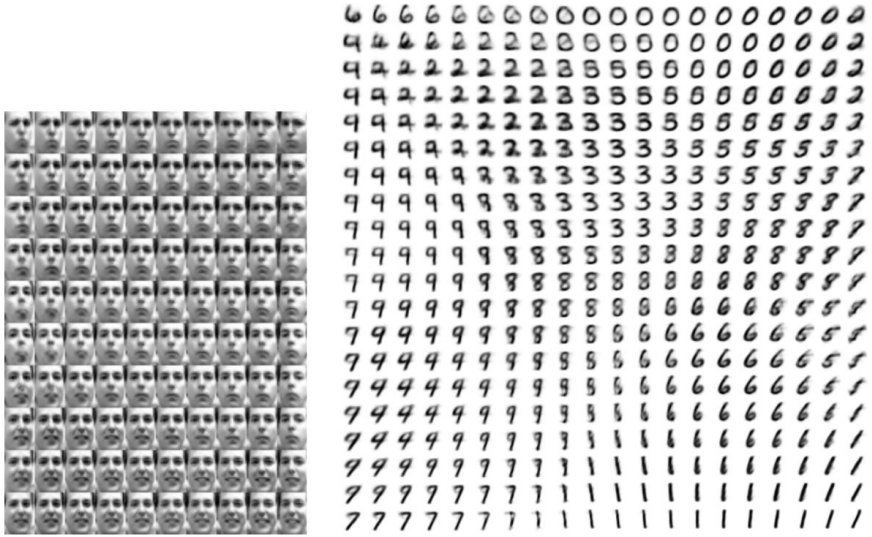


Figure 4: Visualisations of learned data manifold for generative models with two-dimensional latent space, learned with AEVB. Since the prior of the latent space is Gaussian, linearly spaced coordinates on the unit square were transformed through the inverse CDF of the Gaussian to produce values of the latent variables $\mathbf{z}$. For each of these values $\mathbf{z}$, we plotted the corresponding generative $p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})$ with the learned parameters $\boldsymbol{\theta}$.

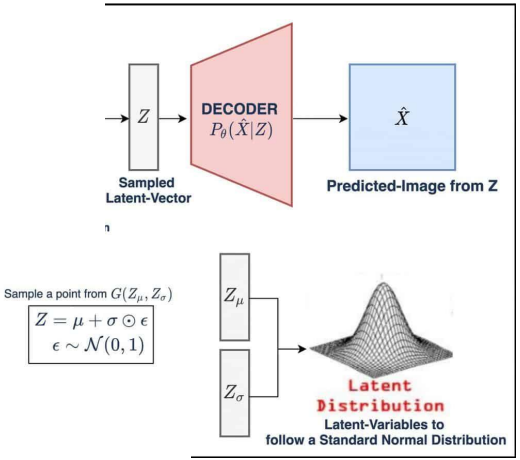
Variational Auto-Encoders

Sample any point from the N-Dimensional Gaussian and render the sample.

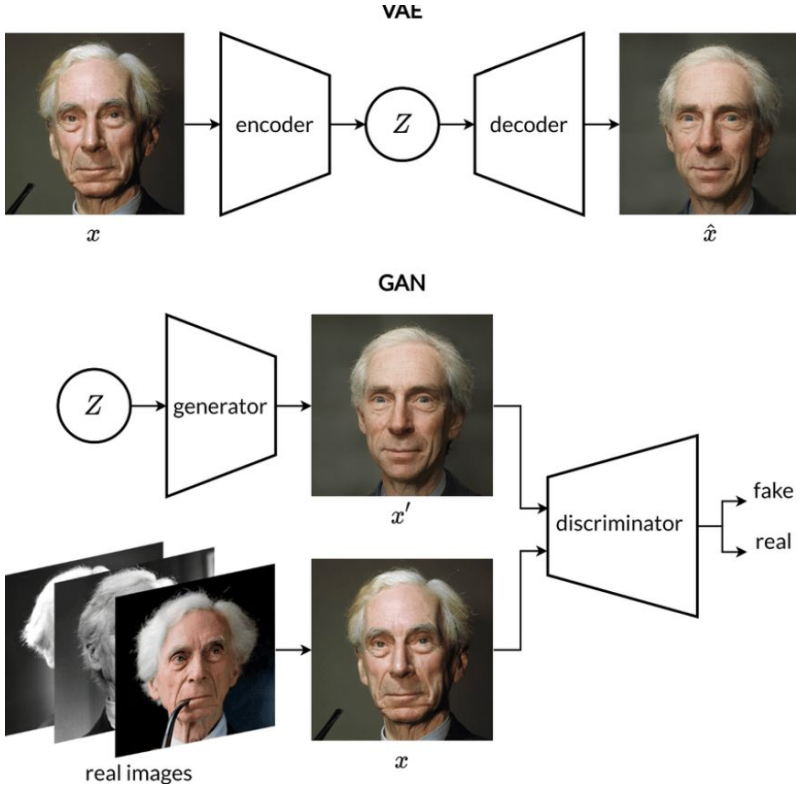


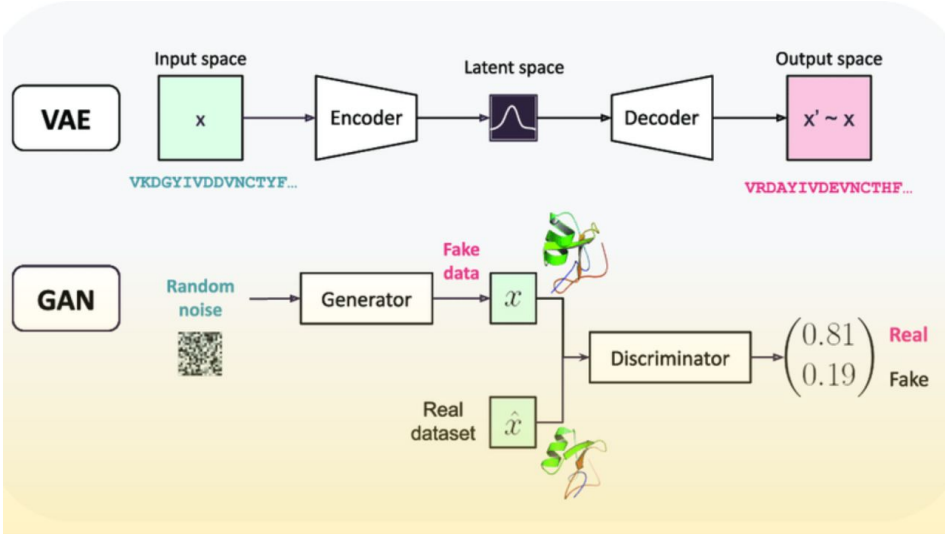
(a) Learned Frey Face manifold (b) Learned MNIST manifold

Figure 4: Visualisations of learned data manifold for generative models with two-dimensional latent space, learned with AEVB. Since the prior of the latent space is Gaussian, linearly spaced coordinates on the unit square were transformed through the inverse CDF of the Gaussian to produce values of the latent variables $\mathbf{z}$. For each of these values $\mathbf{z}$, we plotted the corresponding generative $p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})$ with the learned parameters $\boldsymbol{\theta}$.

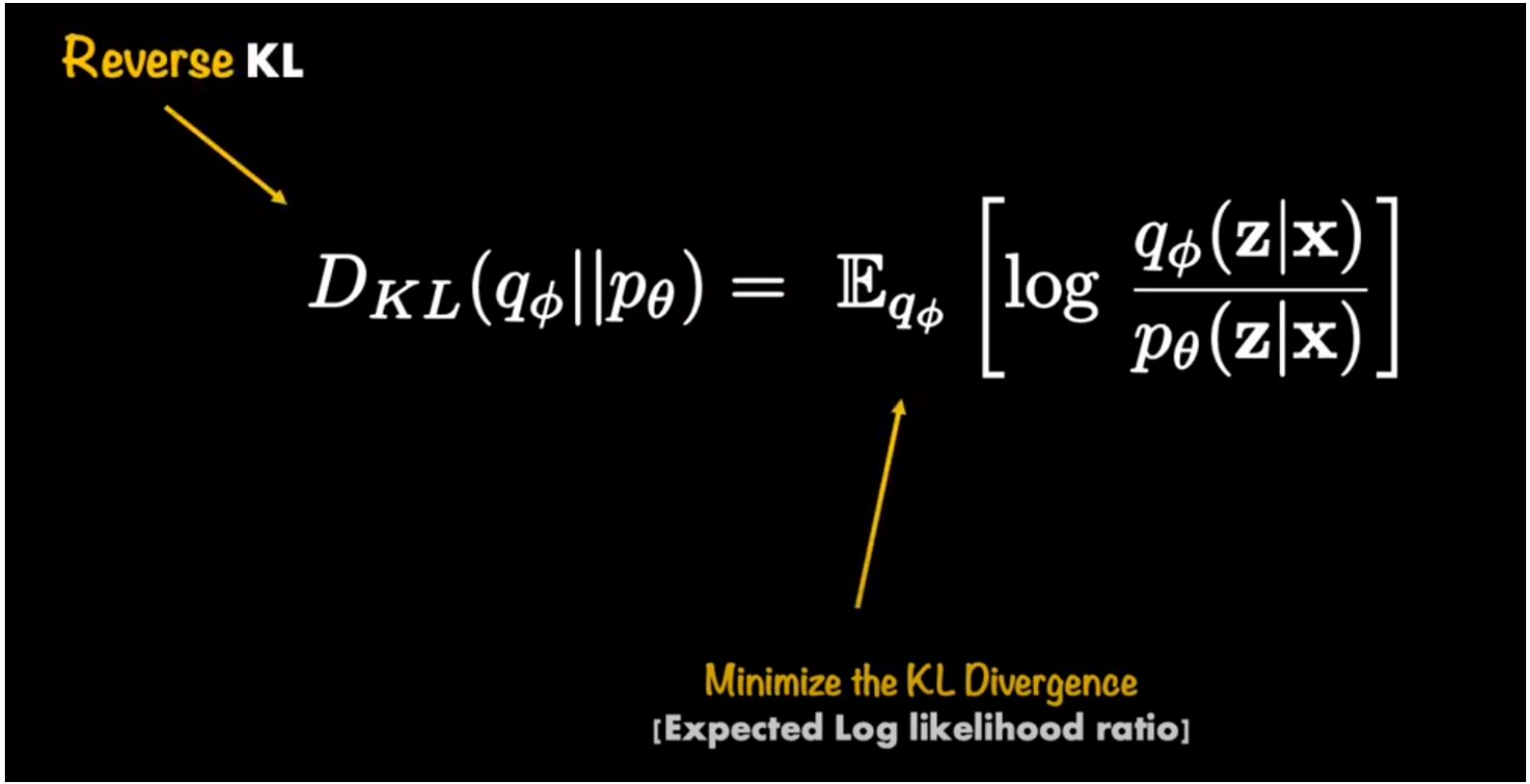


Differences + Similarities between GAN's and VAE's





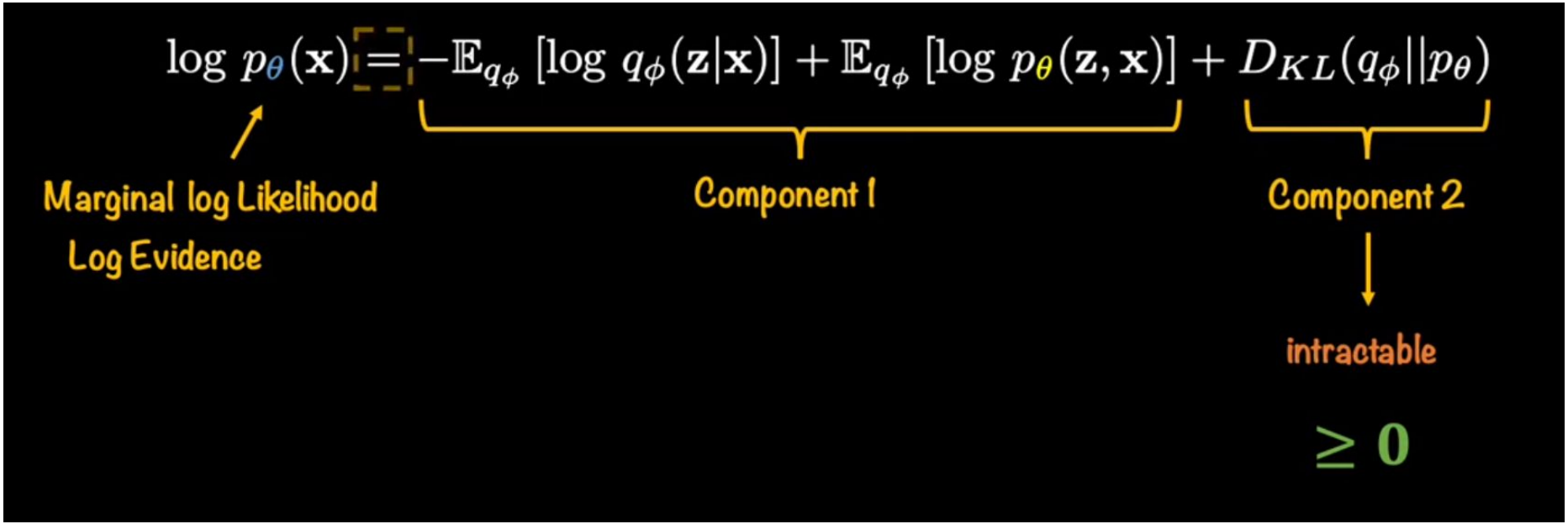
Training the VAE



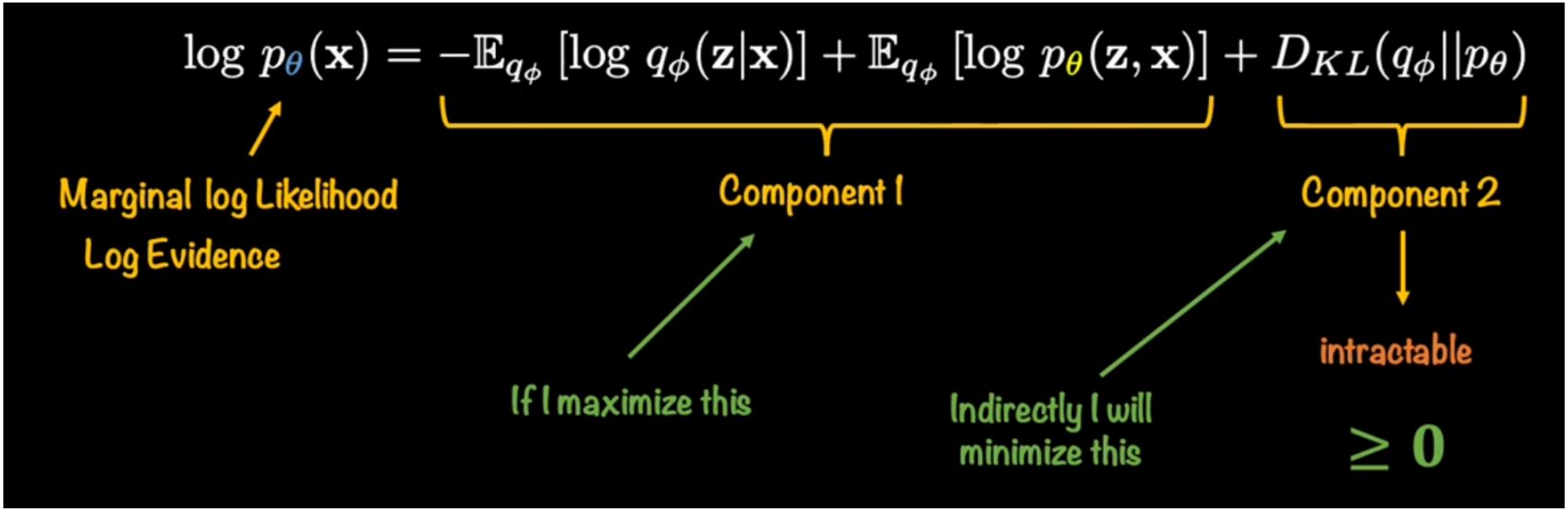
Training the VAE

$$
\begin{aligned}
D_{KL}(q_\phi||p_\theta) &= \mathbb{E}_{q_\phi}\left[\log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_\theta(\mathbf{z}|\mathbf{x})}\right] \\
&= \mathbb{E}_{q_\phi}\left[\log q_\phi(\mathbf{z}|\mathbf{x})\right] - \mathbb{E}_{q_\phi}\left[\log p_\theta(\mathbf{z}|\mathbf{x})\right] \\
&= \mathbb{E}_{q_\phi}\left[\log q_\phi(\mathbf{z}|\mathbf{x})\right] - \mathbb{E}_{q_\phi}\left[\log \frac{p_\theta(\mathbf{z},\mathbf{x})}{p_\theta(\mathbf{x})}\right] \\
&= \mathbb{E}_{q_\phi}\left[\log q_\phi(\mathbf{z}|\mathbf{x})\right] - \mathbb{E}_{q_\phi}\left[\log p_\theta(\mathbf{z},\mathbf{x})\right] + \mathbb{E}_{q_\phi}\left[\log p_\theta(\mathbf{x})\right] \\
&= \mathbb{E}_{q_\phi}\left[\log q_\phi(\mathbf{z}|\mathbf{x})\right] - \mathbb{E}_{q_\phi}\left[\log p_\theta(\mathbf{z},\mathbf{x})\right] + \int q_\phi(\mathbf{z}|\mathbf{x}) \log p_\theta(\mathbf{x})\, d\mathbf{z} \\
&= \mathbb{E}_{q_\phi}\left[\log q_\phi(\mathbf{z}|\mathbf{x})\right] - \mathbb{E}_{q_\phi}\left[\log p_\theta(\mathbf{z},\mathbf{x})\right] + \log p_\theta(\mathbf{x}) \int q_\phi(\mathbf{z}|\mathbf{x})\, d\mathbf{z} \\
&= \mathbb{E}_{q_\phi}\left[\log q_\phi(\mathbf{z}|\mathbf{x})\right] - \mathbb{E}_{q_\phi}\left[\log p_\theta(\mathbf{z},\mathbf{x})\right] + \log p_\theta(\mathbf{x})
\end{aligned}
$$

Training the VAE



Training the VAE



Training the VAE

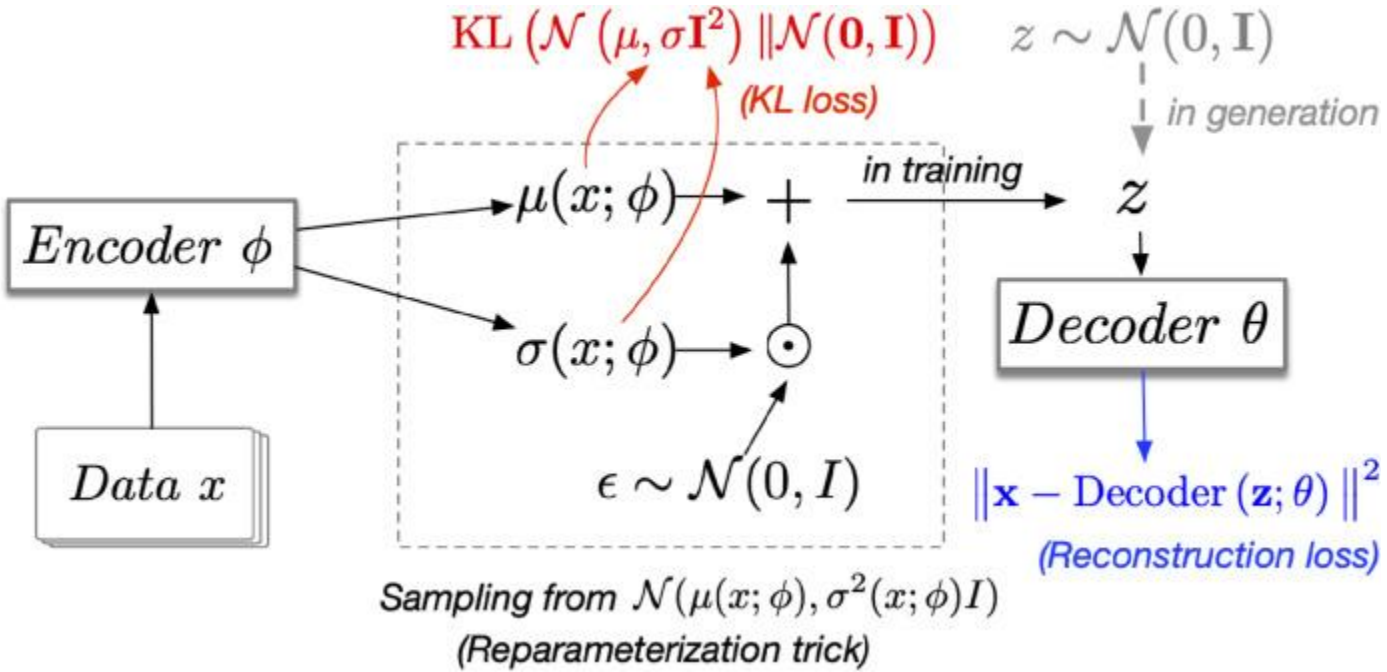
$$
\begin{aligned}
\text{ELBO} &= -\mathbb{E}_{q_\phi}\left[\log q_\phi(\mathbf{z}|\mathbf{x})\right] + \mathbb{E}_{q_\phi}\left[\log p_\theta(\mathbf{z},\mathbf{x})\right] \\
&= -\mathbb{E}_{q_\phi}\left[\log q_\phi(\mathbf{z}|\mathbf{x})\right] + \mathbb{E}_{q_\phi}\left[\log p_\theta(\mathbf{x}|\mathbf{z})\right] + \mathbb{E}_{q_\phi}\left[\log p_\theta(\mathbf{z})\right] \\
&= \mathbb{E}_{q_\phi}\left[\log p_\theta(\mathbf{x}|\mathbf{z})\right] - \mathbb{E}_{q_\phi}\left[\log q_\phi(\mathbf{z}|\mathbf{x})\right] + \mathbb{E}_{q_\phi}\left[\log p_\theta(\mathbf{z})\right] \\
&= \mathbb{E}_{q_\phi}\left[\log p_\theta(\mathbf{x}|\mathbf{z})\right] - \mathbb{E}_{q_\phi}\left[\log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_\theta(\mathbf{z})}\right]
\end{aligned}
$$

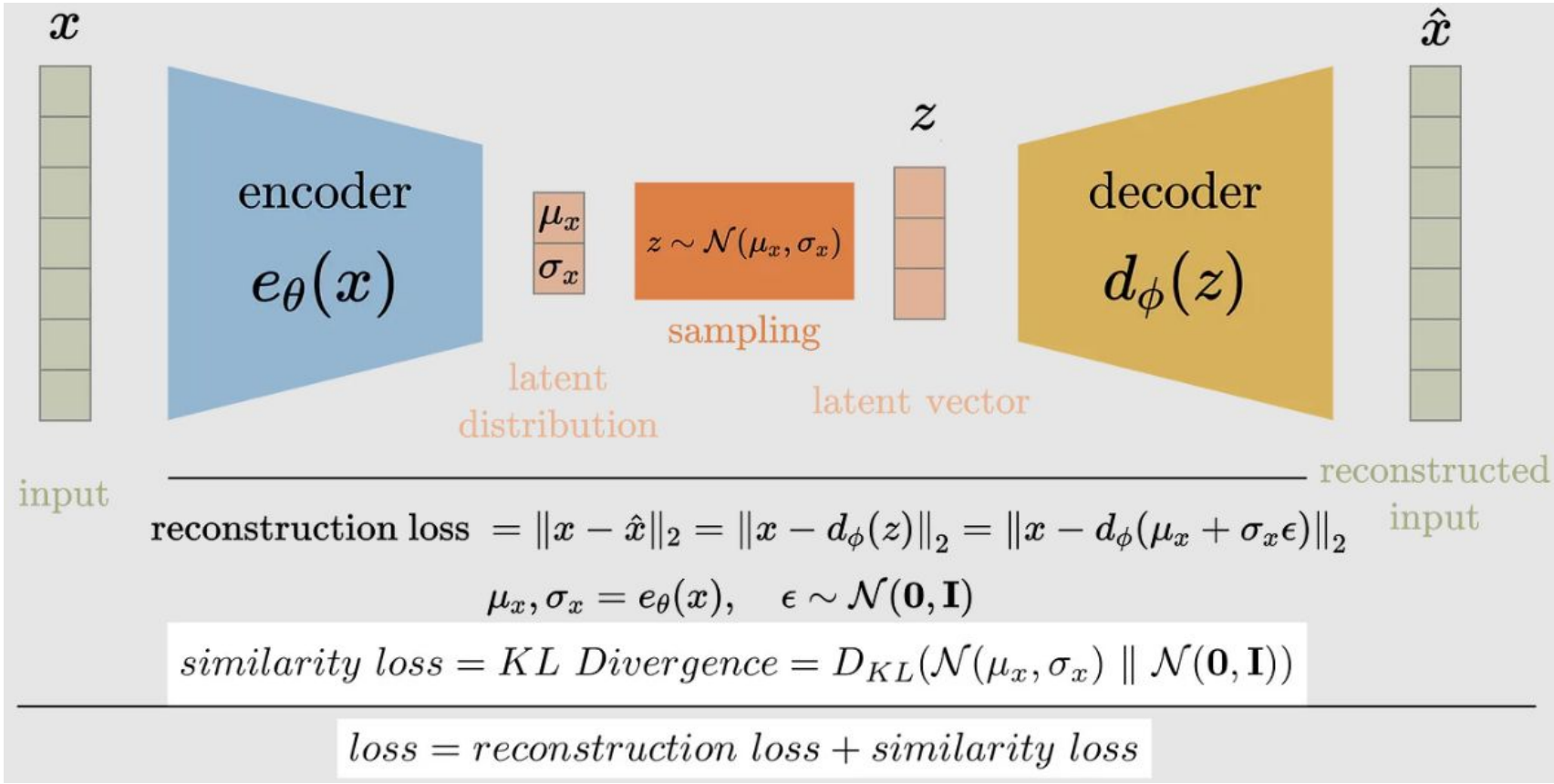
Training the VAE

$$\text{ELBO} = \mathbb{E}_{q_\phi}\left[\log p_\theta(\mathbf{x}|\mathbf{z})\right] - \mathbb{E}_{q_\phi}\left[\log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_\theta(\mathbf{z})}\right]$$

Expected reconstruction error

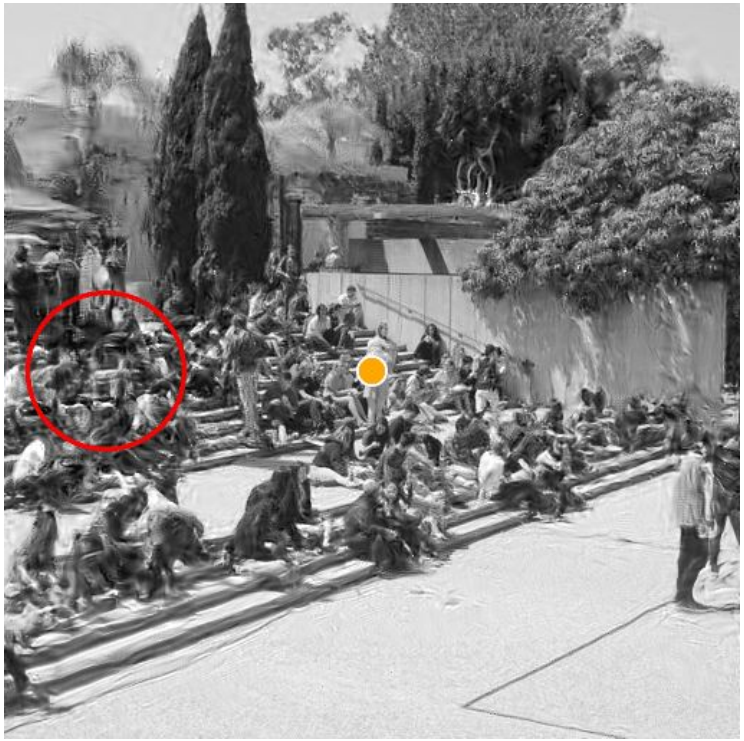
KL Divergence between approx. posterior & the prior


$\mathrm{KL}\left(\mathcal{N}\left(\mu, \sigma\mathbf{I}^2\right) \| \mathcal{N}(\mathbf{0}, \mathbf{I})\right)$
(KL loss)
$z \sim \mathcal{N}(0, \mathrm{I})$
in generation
in training
Encoder ϕ
$\mu(x; \phi)$
$\sigma(x; \phi)$
$+$
$\odot$
z
Decoder θ
Data x
$\epsilon \sim \mathcal{N}(0, I)$
$\|\mathbf{x} - \mathrm{Decoder}\,(\mathbf{z}; \theta)\|^2$
(Reconstruction loss)
Sampling from $\mathcal{N}(\mu(x; \phi), \sigma^2(x; \phi)I)$
(Reparameterization trick)


x
encoder
$e_\theta(x)$
μ_x
σ_x
latent distribution
$z \sim \mathcal{N}(\mu_x, \sigma_x)$
sampling
z
latent vector
decoder
$d_\phi(z)$
$\hat{x}$
input
reconstructed input
reconstruction loss $= \|x - \hat{x}\|_2 = \|x - d_\phi(z)\|_2 = \|x - d_\phi(\mu_x + \sigma_x \epsilon)\|_2$
$\mu_x, \sigma_x = e_\theta(x), \quad \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
$similarity\ loss = KL\ Divergence = D_{KL}(\mathcal{N}(\mu_x, \sigma_x) \parallel \mathcal{N}(\mathbf{0}, \mathbf{I}))$
$loss = reconstruction\ loss + similarity\ loss$

Towards Metamerism via Foveated Style Transfer

Arturo Deza[1,4], **Aditya Jonnalagadda**[3], **Miguel P. Eckstein**[1,2,4]
[1] Dynamical Neuroscience, [2]Psychological and Brain Sciences,
[3]Electric and Computer Engineering, [4] Institute for Collaborative Biotechnologies
UC Santa Barbara, CA, USA
deza@dyns.ucsb.edu,aditya_jonnalagada@ece.ucsb.edu,eckstein@psych.ucsb.edu



Abstract

The problem of *visual metamerism* is defined as finding a family of perceptually indistinguishable, yet physically different images. In this paper, we propose our NeuroFovea metamer model, a foveated generative model that is based on a mixture of peripheral representations and style transfer forward-pass algorithms. Our gradient-descent free model is parametrized by a foveated VGG19 encoder-decoder which allows us to encode images in high dimensional space and interpolate between the content and texture information with adaptive instance normalization anywhere in the visual field. Our contributions include: 1) A framework for computing metamers that resembles a noisy communication system via a foveated feed-forward encoder-decoder network – We observe that metamerism arises as a byproduct of noisy perturbations that partially lie in the perceptual null space; 2) A perceptual optimization scheme as a solution to the hyperparametric nature of our metamer model that requires tuning of the image-texture tradeoff coefficients everywhere in the visual field which are a consequence of internal noise; 3) An ABX psychophysical evaluation of our metamers where we also find that the rate of growth of the receptive fields in our model match V1 for reference metamers and V2 between synthesized samples. Our model also renders metamers at roughly a second, presenting a $\times 1000$ speed-up compared to the previous work, which allows for tractable data-driven metamer experiments.

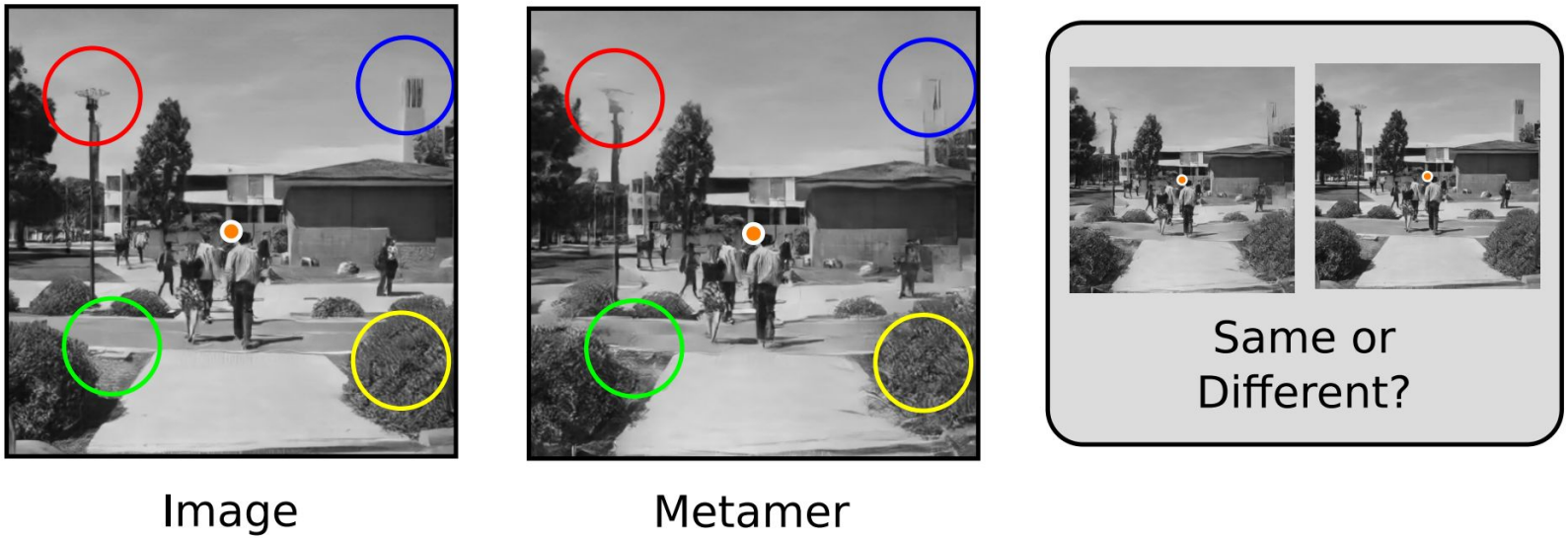


Figure 1: Two visual metamers are physically different images that when fixated on the orange dot (center), should remain perceptually indistinguishable to each other for an observer. Colored circles highlight different distortions in the visual field that observers do not perceive in our model.

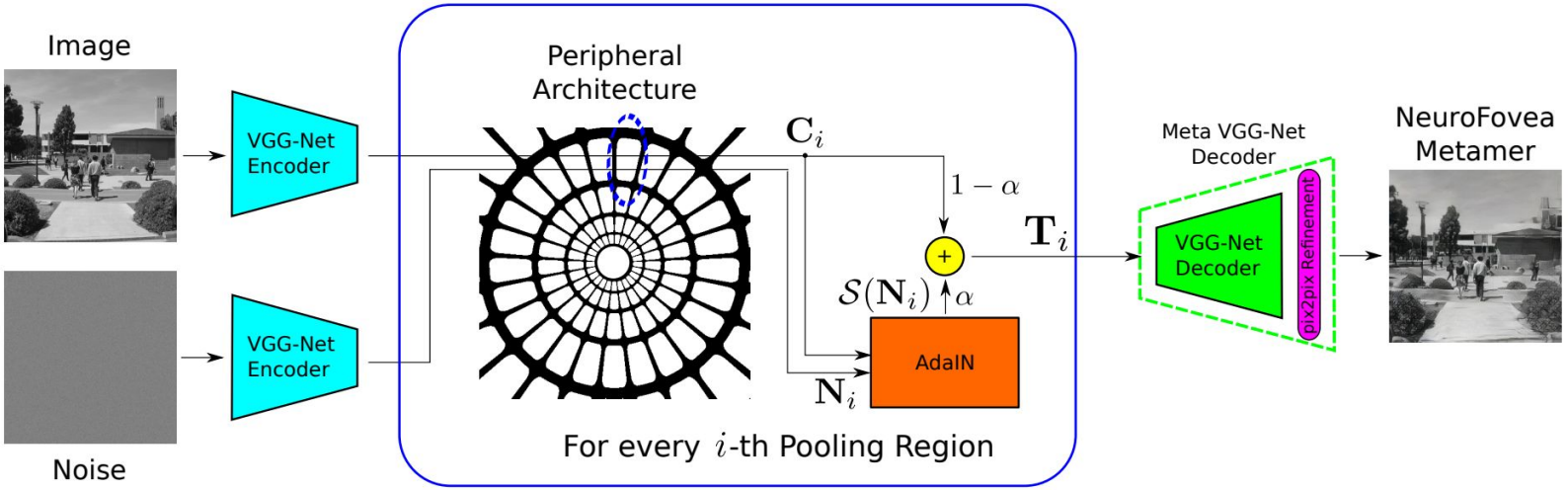
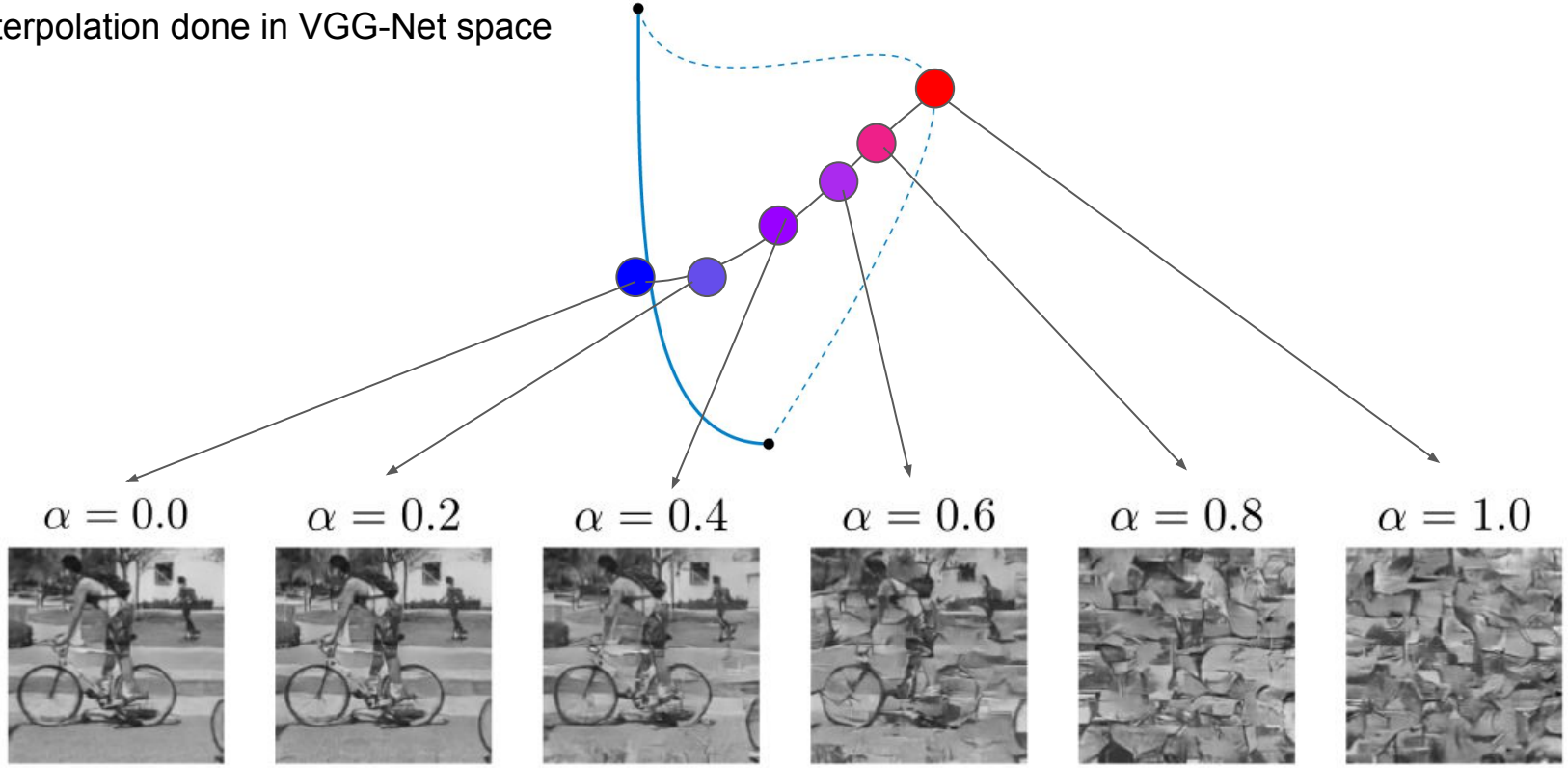


Figure 2: The NeuroFovea metamer generation schematic: An input image and a noise patch are fed through a VGG-Net encoder into a new feature space. Through spatial control we can produce an interpolation for each pooling region in such feature space between the stylized-noise (texture), and the content (the input image). This is how we successfully impose both global image and local texture-like constraints in every pooling region. The metamer is the output of the pooled (and interpolated) feature vector through the Meta VGG-Net Decoder.

$$\mathbf{M}(I|\mathcal{N};\bar{\alpha}) = \mathcal{D}(\mathcal{E}_{\sum}(I|\mathcal{N};\bar{\alpha})) = \mathcal{D}(\sum_{i=1}^{k} w_i[(1-\alpha_i)\mathcal{E}_i(I) + \alpha_i \mathcal{S}(\mathcal{E}_i(\mathcal{N}))])$$

Interpolation done in VGG-Net space



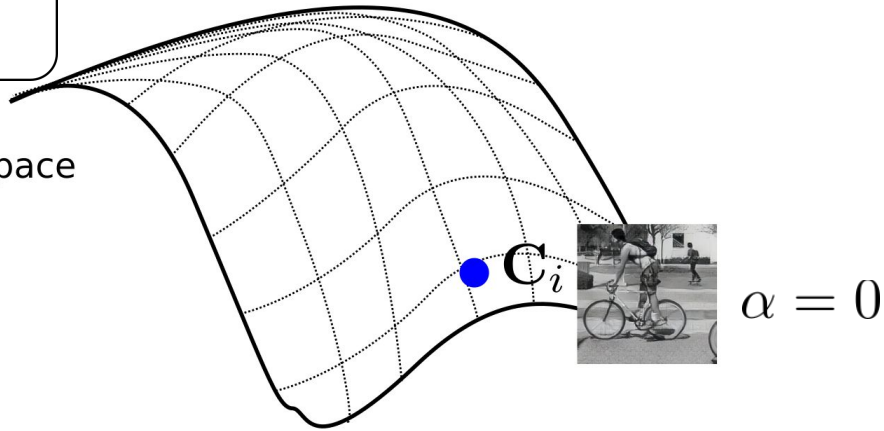
Case Study:

What is the maximum amount of texture-driven distortion we can put in this receptive field?

Case Study:

What is the maximum amount of texture-driven distortion we can put in this receptive field?

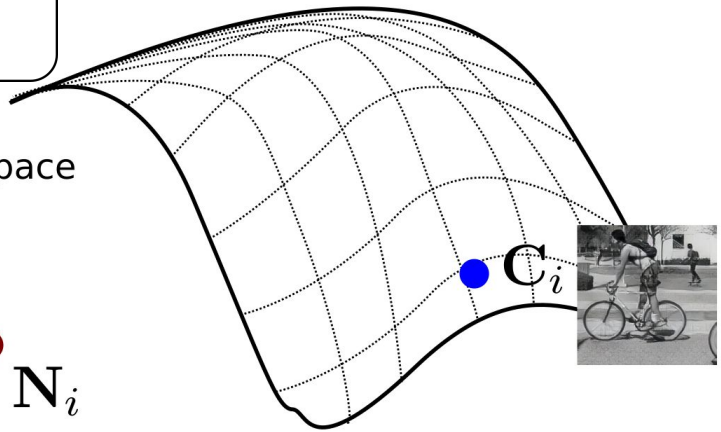
Encoded Space



Case Study:

What is the maximum amount of texture-driven distortion we can put in this receptive field?

Encoded Space



Case Study:

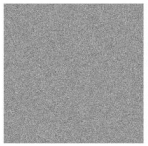
What is the maximum amount of texture-driven distortion we can put in this receptive field?

$\alpha = 1$

$\mathcal{S}(\mathbf{N}_i)$

Encoded Space

$\mathcal{S}(\circ)$



$\mathbf{N}_i$

$\mathbf{C}_i$

$\alpha = 0$

Case Study:

What is the maximum amount of texture-driven distortion we can put in this receptive field?

$\alpha = 1$

$\mathcal{S}(\mathbf{N}_i)$

$\alpha = 0.4$

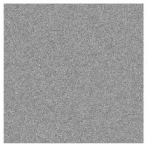
Encoded Space

$\mathbf{T}_i$

$\mathcal{S}(\circ)$

$\mathbf{C}_i$

$\alpha = 0$



$\mathbf{N}_i$

Case Study:

What is the maximum amount of texture-driven distortion we can put in this receptive field?

$\alpha = 1$

$\mathcal{S}(\mathbf{N}_i)$

$\alpha = 0.4$

Encoded Space

$\mathcal{S}^{\perp}(\mathbf{N}_i)$

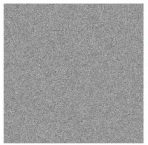
$\mathbf{T}_i$

$\mathcal{S}(\circ)$

$\mathcal{S}^{\parallel}(\mathbf{N}_i)$

$\mathbf{C}_i$

$\alpha = 0$

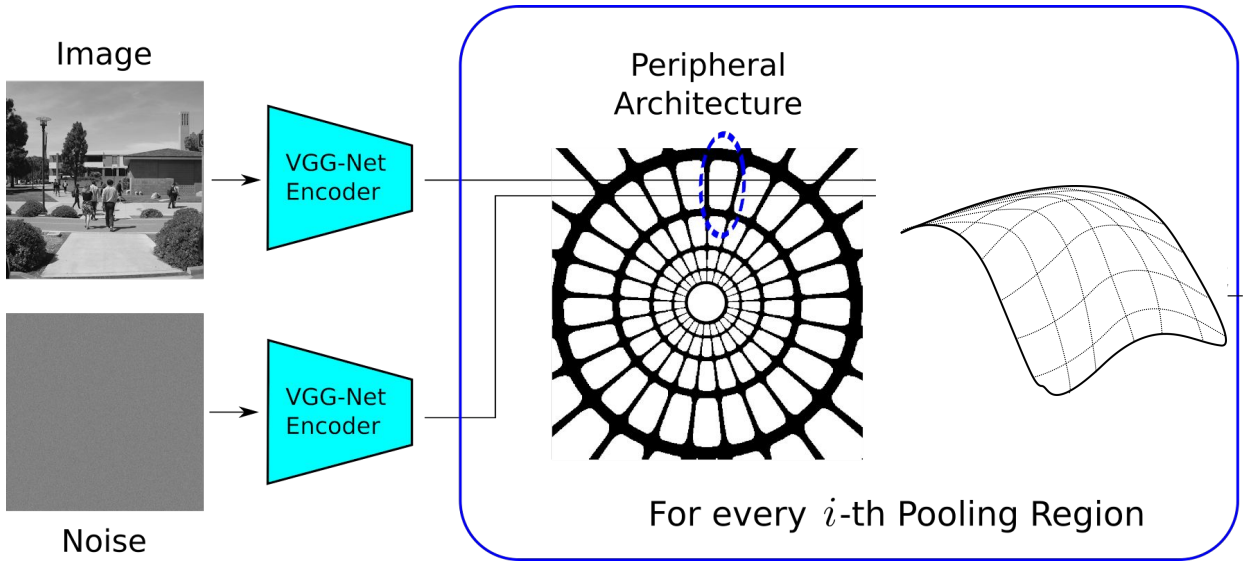


$\mathbf{N}_i$

P

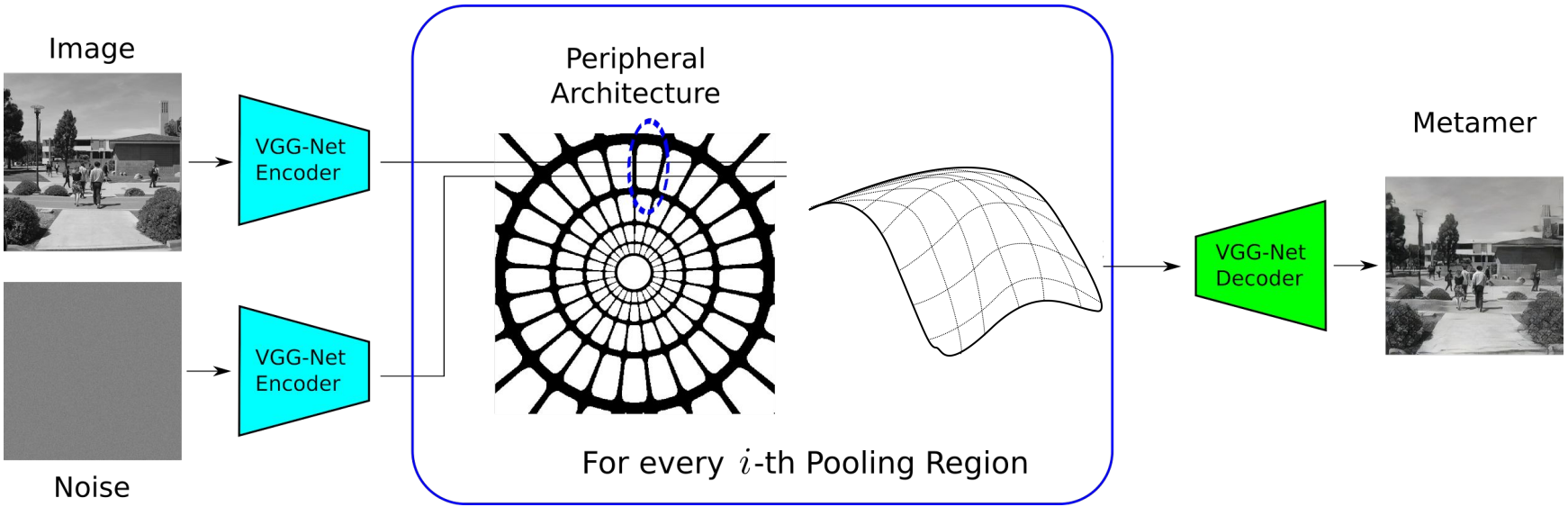
Perceptual Space

A new metamer rendering pipeline



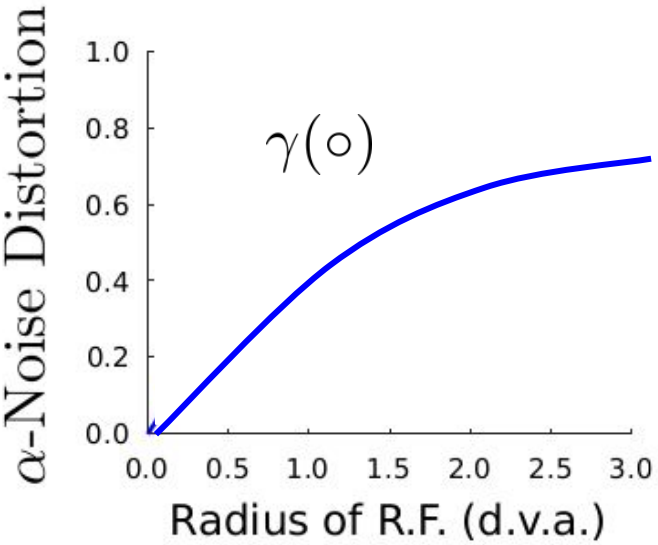
Deza, Jonnalagadda & Eckstein. ICLR 2019.

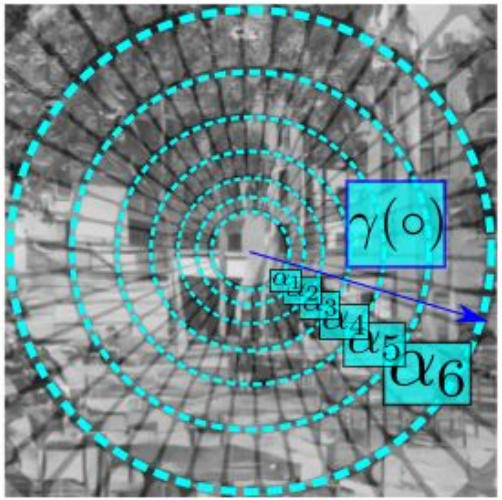
A new metamer rendering pipeline



Deza, Jonnalagadda & Eckstein. ICLR 2019.

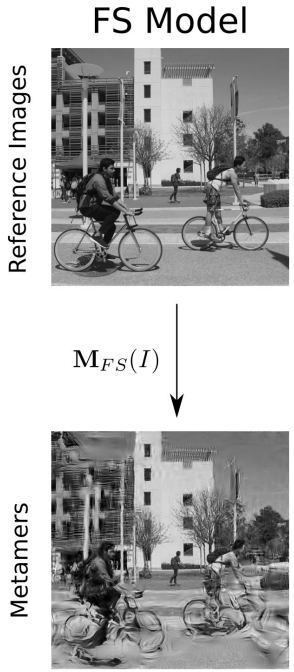
How much can we distort in each pooling region?





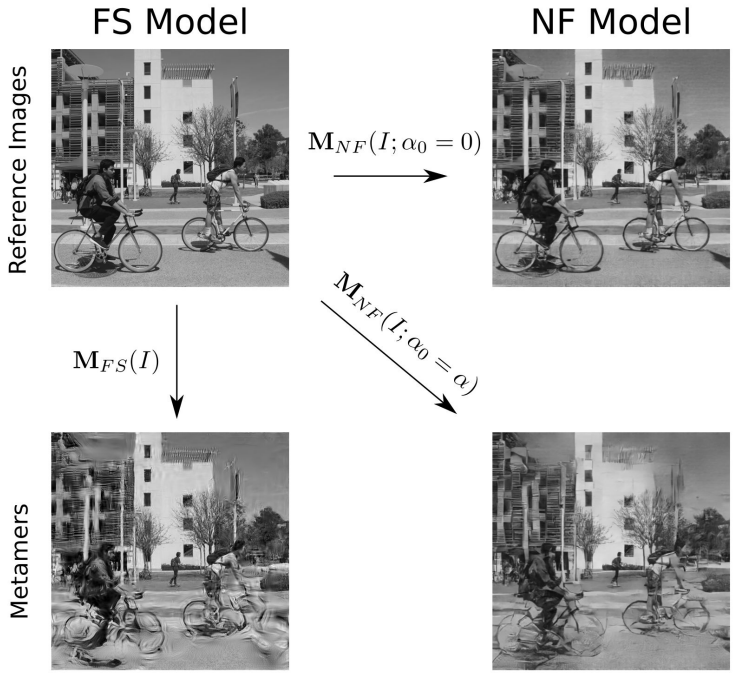
Deza, Jonnalagadda & Eckstein. ICLR 2019.

Perceptual Optimization Procedure



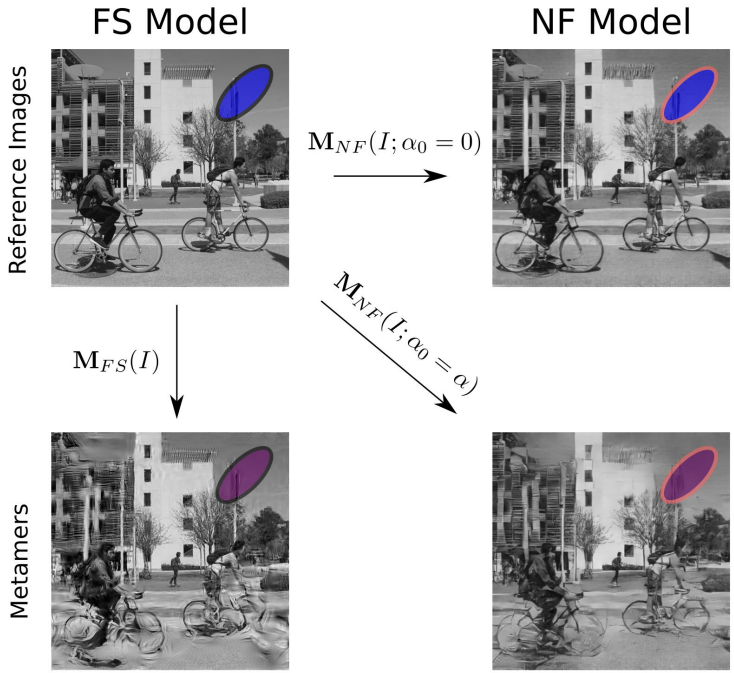
Deza, Jonnalagadda & Eckstein. ICLR 2019.

Perceptual Optimization Procedure



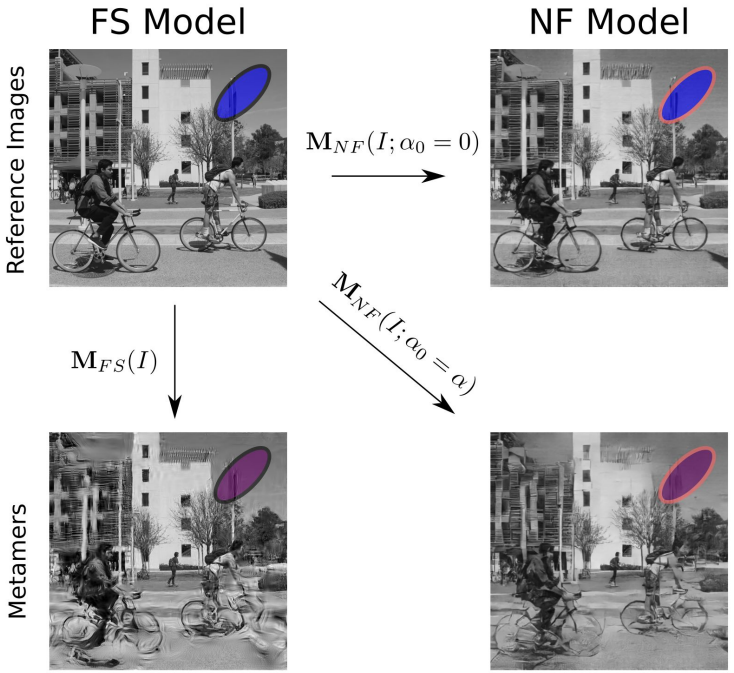
Deza, Jonnalagadda & Eckstein. ICLR 2019.

Perceptual Optimization Procedure



Deza, Jonnalagadda & Eckstein. ICLR 2019.

Perceptual Optimization Procedure



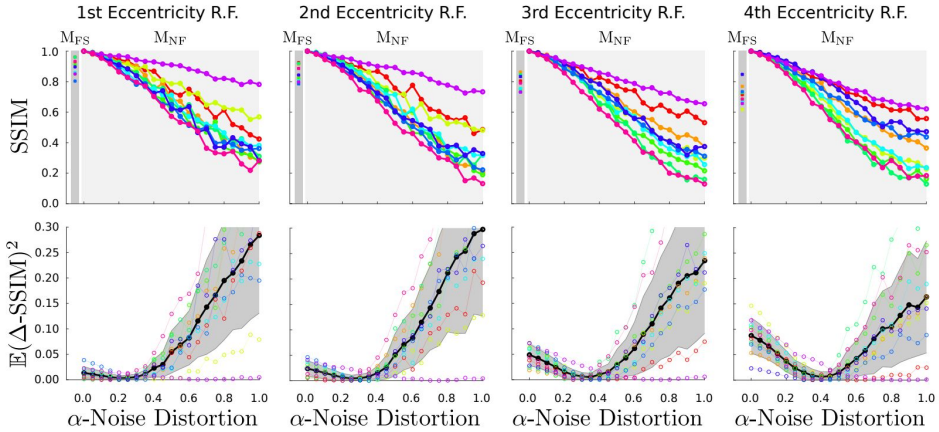
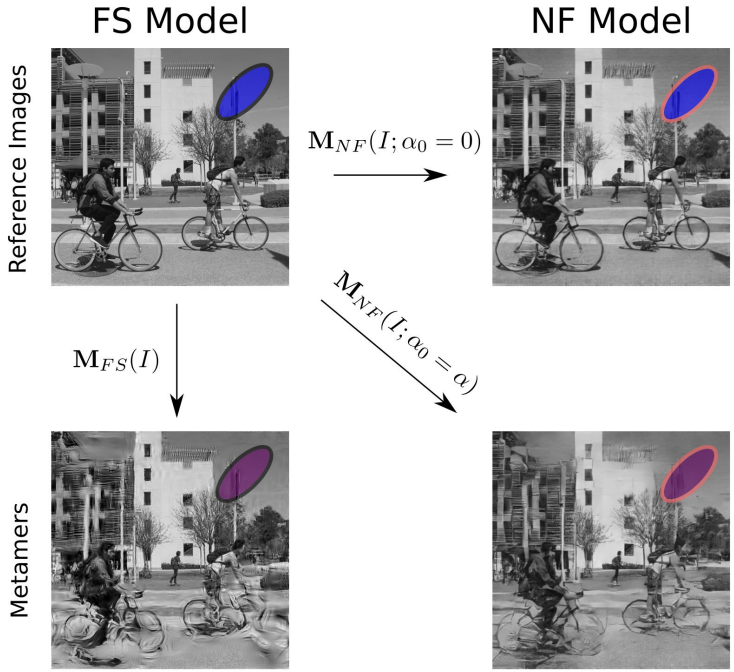


Figure 8: The result of each SSIM (top) for Experiment 1 for a scale of $s = 0.3$ where we find the critical α for each receptive field ring as we minimize $\mathbb{E}(\Delta\text{-SSIM})^2$ (bottom). $\mathbb{E}(\Delta\text{-SSIM})^2$ is minimized by matching the perceptual distortion of the Freeman & Simoncelli (2011) (M_{FS}) and NeuroFovea (M_{NF}) metamers in Eq. 9. Each color represents a different 512×512 image trajectory, the black line (bottom) shows the average. Only the first 4 eccentricity dependent receptive fields are shown.

Deza, Jonnalagadda & Eckstein. ICLR 2019.

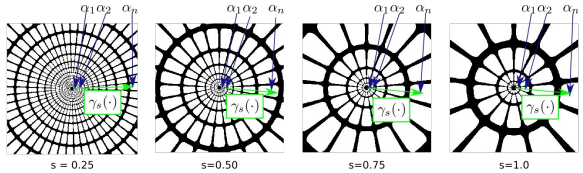
Perceptual Optimization Procedure



Minimize : $(\text{SSIM}(\,,\,) - \text{SSIM}(\,,\,))^2$

for every pooling region in the visual field

Estimation of maximum distortion in the visual field



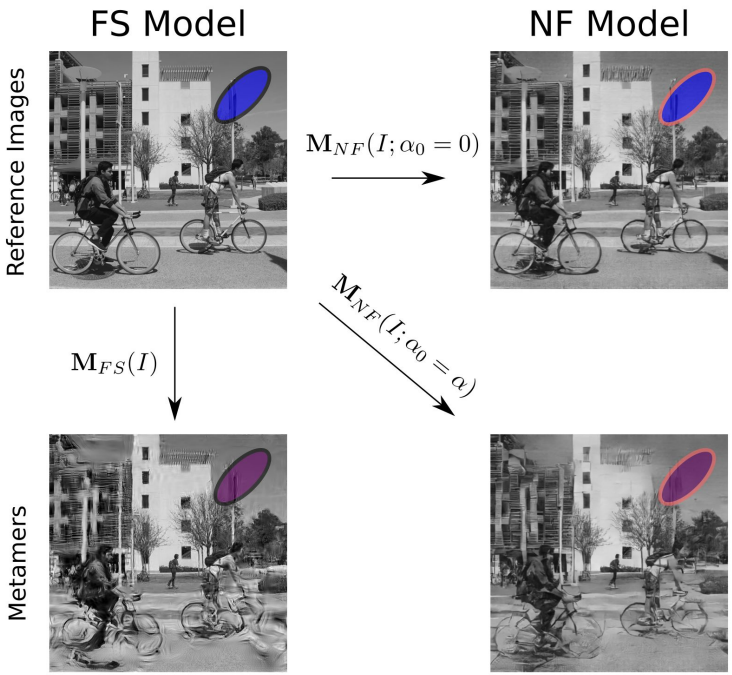
$$s_0, \bar{\alpha_0} = \text{argmax}_{s,\bar{\alpha}} \mathbb{E}[d'(s, \bar{\alpha}|\theta_{obs})]$$
$$\text{s.t. } 0 < d'(s, \bar{\alpha}|\theta_{obs}) < \epsilon$$

$$\downarrow \alpha = \gamma(\circ; s)$$

$$s_0 = \text{argmax}_s \, \mathbb{E}[d'(s, \gamma(\circ; s)|\theta_{obs})]$$
$$\text{s.t. } 0 < d'(s, \gamma(\circ; s)|\theta_{obs}) < \epsilon$$

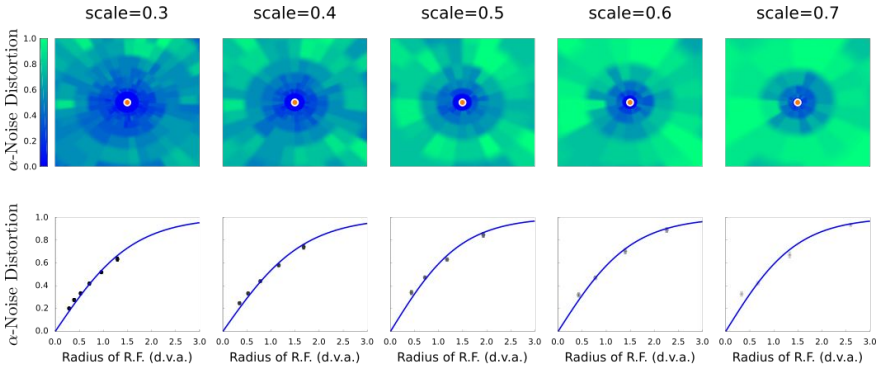
Deza, Jonnalagadda & Eckstein. ICLR 2019.

Perceptual Optimization Procedure



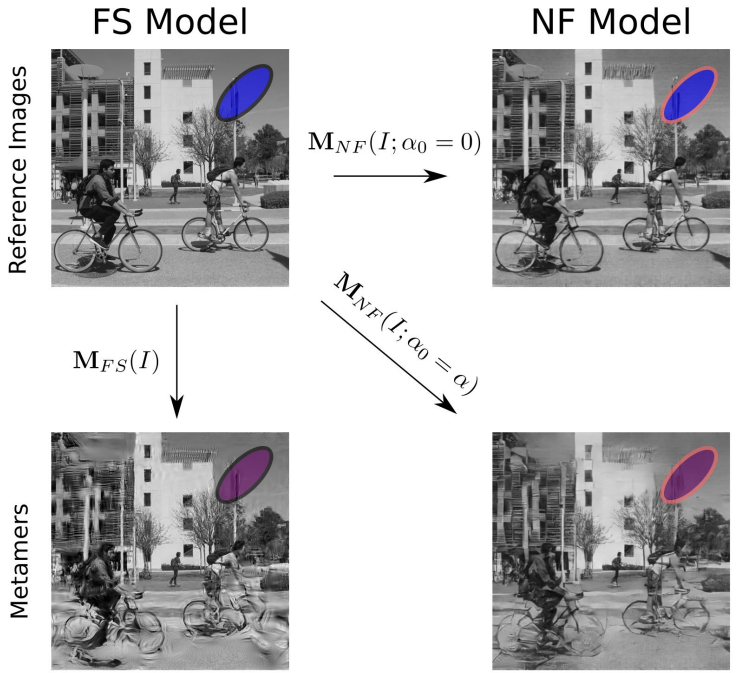
Minimize : $(\text{SSIM}(\,,\,) - \text{SSIM}(\,,\,))^2$

for every pooling region in the visual field



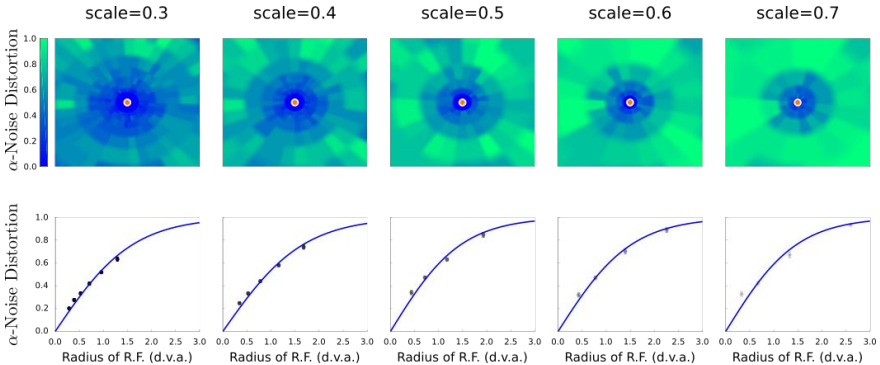
Deza, Jonnalagadda & Eckstein. ICLR 2019.

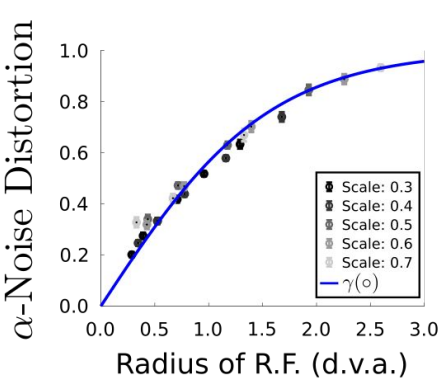
Perceptual Optimization Procedure

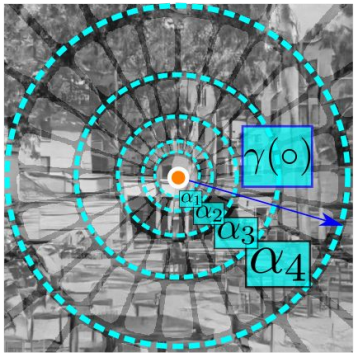


Minimize : $(\text{SSIM}(\,,\,) - \text{SSIM}(\,,\,))^2$

for every pooling region in the visual field



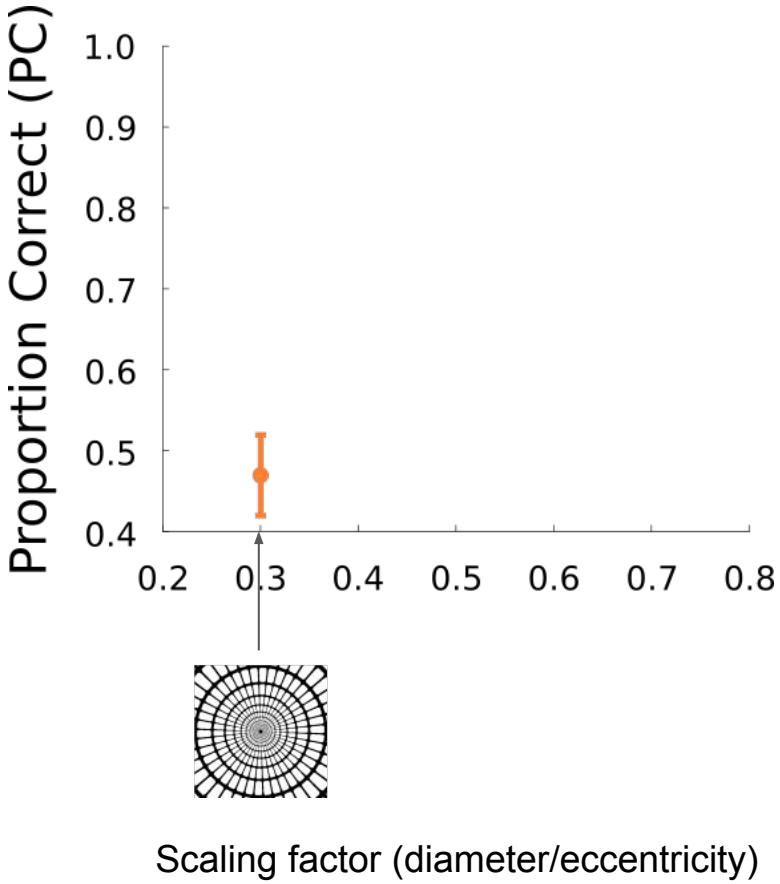




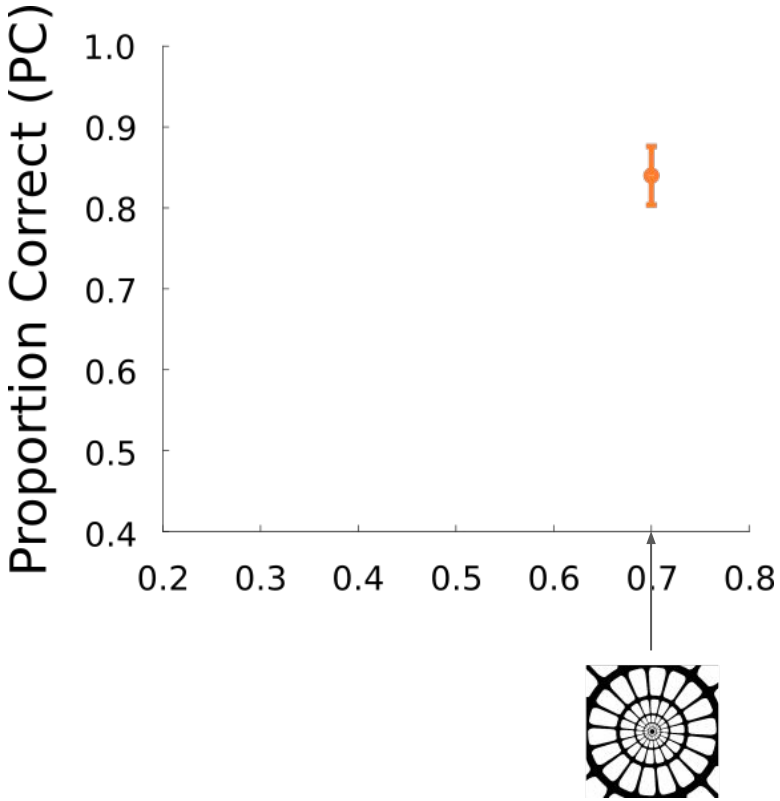
scale = 0.5

Deza, Jonnalagadda & Eckstein. ICLR 2019.

How do we test this?

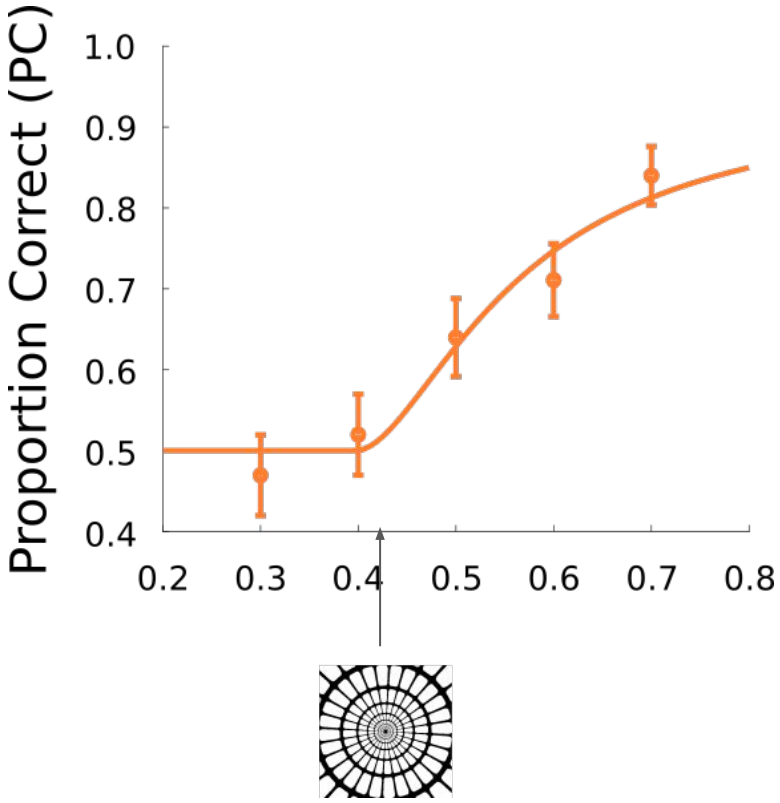


How do we test this?



Scaling factor (diameter/eccentricity)

How do we test this?



Scaling factor (diameter/eccentricity)

Fundamentals of Rate-Distortion Theory

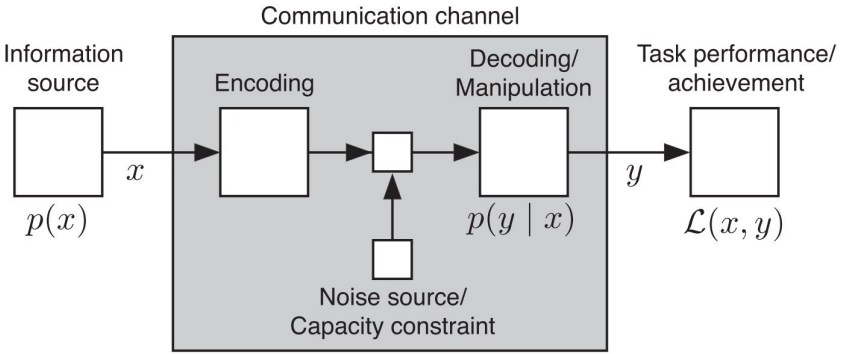


Fig. 1. The core constructs of rate–distortion theory. An information source is described by a probability distribution over its alphabet. Samples from this source are communicated over a noisy or capacity-limited channel, resulting in a conditional probability distribution over the channel output. A cost function defines the consequences of error in communication.

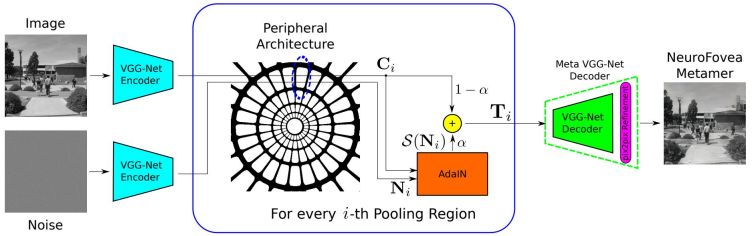
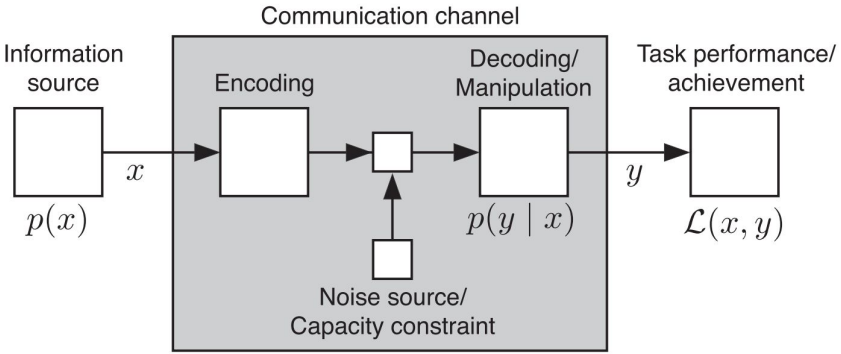


Figure 2: The NeuroFovea metamer generation schematic: An input image and a noise patch are fed through a VGG-Net encoder into a new feature space. Through spatial control we can produce an interpolation for each pooling region in such feature space between the stylized-noise (texture), and the content (the input image). This is how we successfully impose both global image and local texture-like constraints in every pooling region. The metamer is the output of the pooled (and interpolated) feature vector through the Meta VGG-Net Decoder.

Sims. Cognition, 2016.

Fundamentals of Rate-Distortion Theory



$p(x)$: Statistics of information source

$p(y \mid x)$: Capacity-limited information channel

$\mathcal{L}(x, y)$: Cost of communication error

Fig. 1. The core constructs of rate–distortion theory. An information source is described by a probability distribution over its alphabet. Samples from this source are communicated over a noisy or capacity-limited channel, resulting in a conditional probability distribution over the channel output. A cost function defines the consequences of error in communication.

Sims. Cognition, 2016.

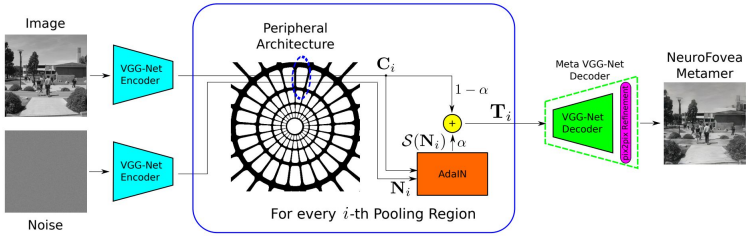
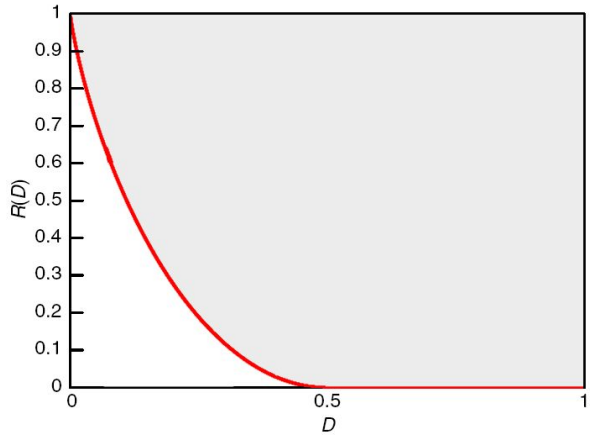


Figure 2: The NeuroFovea metamer generation schematic: An input image and a noise patch are fed through a VGG-Net encoder into a new feature space. Through spatial control we can produce an interpolation for each pooling region in such feature space between the stylized-noise (texture), and the content (the input image). This is how we successfully impose both global image and local texture-like constraints in every pooling region. The metamer is the output of the pooled (and interpolated) feature vector through the Meta VGG-Net Decoder.

Example of a RD curve:



β-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework

Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, Alexander Lerchner
Google DeepMind
{irinah,lmatthey,arkap,cpburgess,glorotx,
botvinick,shakir,lerchner}@google.com

Abstract

Learning an interpretable factorised representation of the independent data generative factors of the world without supervision is an important precursor for the development of artificial intelligence that is able to learn and reason in the same way that humans do. We introduce β-VAE, a new state-of-the-art framework for automated discovery of interpretable factorised latent representations from raw image data in a completely unsupervised manner. Our approach is a modification of the variational autoencoder (VAE) framework. We introduce an adjustable hyperparameter β that balances latent channel capacity and independence constraints with reconstruction accuracy. We demonstrate that β-VAE with appropriately tuned $\beta > 1$ qualitatively outperforms VAE ($\beta = 1$), as well as state of the art unsupervised (InfoGAN) and semi-supervised (DC-IGN) approaches to disentangled factor learning on a variety of datasets (*celebA*, *faces* and *chairs*). Furthermore, we devise a protocol to quantitatively compare the degree of disentanglement learnt by different models, and show that our approach also significantly outperforms all baselines quantitatively. Unlike InfoGAN, β-VAE is stable to train, makes few assumptions about the data and relies on tuning a single hyperparameter β, which can be directly optimised through a hyperparameter search using weakly labelled data or through heuristic visual inspection for purely unsupervised data.

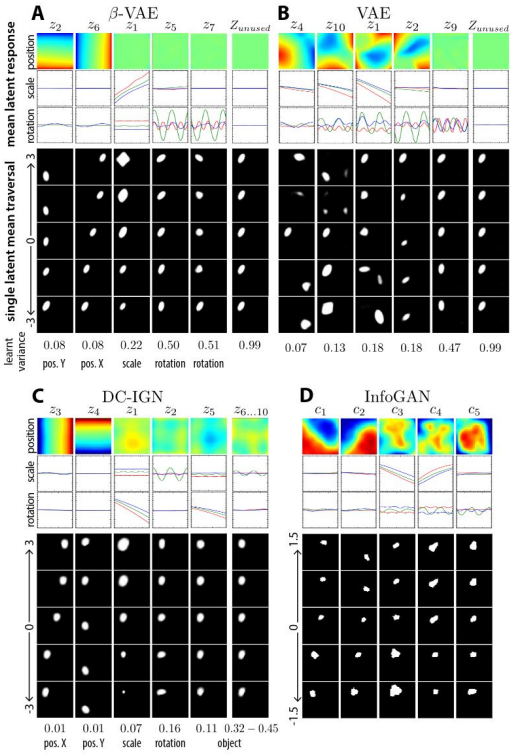
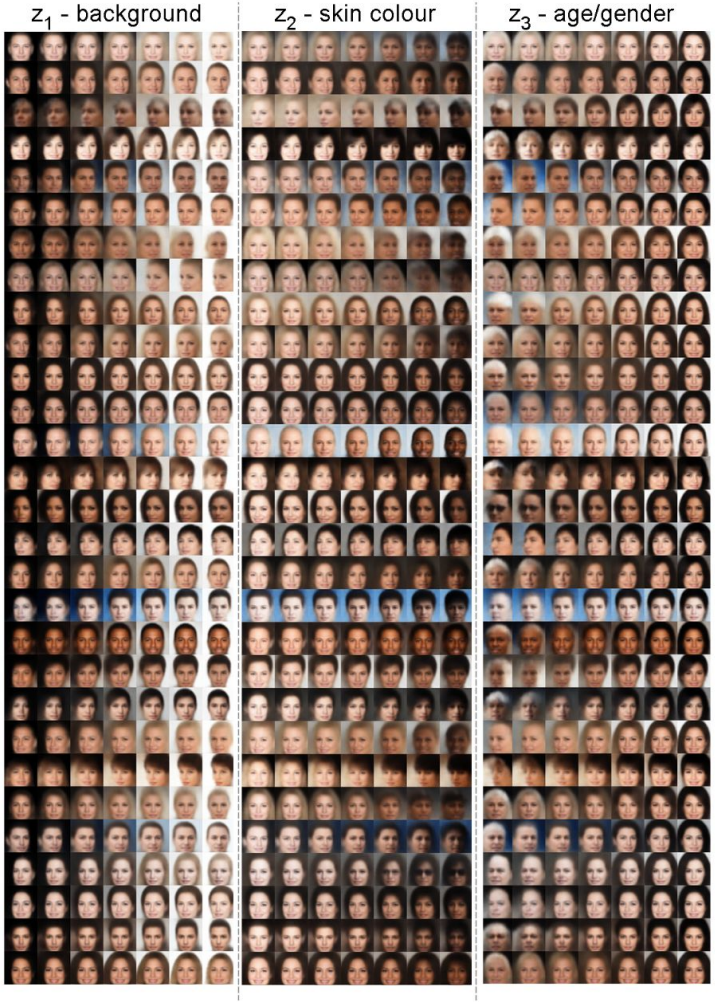


Figure 7: **A:** Representations learnt by a β-VAE ($\beta = 4$). Each column represents a latent z_i, ordered according to the learnt Gaussian variance (last row). Row 1 (position) shows the mean activation (red represents high values) of each latent z_i as a function of all 32x32 locations averaged across objects, rotations and scales. Row 2 and 3 show the mean activation of each unit z_i as a function of scale (respectively rotation), averaged across rotations and positions (respectively scales and positions). *Square* is red, *oval* is green and *heart* is blue. Rows 4-8 (second group) show reconstructions resulting from the traversal of each latent z_i over three standard deviations around the unit Gaussian prior mean while keeping the remaining 9/10 latent units fixed to the values obtained by running inference on an image from the dataset. **B:** Similar analysis for VAE ($\beta = 1$). **C:** Similar analysis for DC-IGN, clamping a single latent each for scale, positions, orientation and 5 for shape. **D:** Similar analysis for InfoGAN, using 5 continuous latents regularized using the mutual information cost, and 5 additional unconstrained noise latents (not shown).



$$\mathcal{F}(\theta, \phi, \beta; \mathbf{x}, \mathbf{z}) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x} \mid \mathbf{z})] - \beta[D_{KL}(\log q_\theta(\mathbf{z} \mid \mathbf{x})||p(\mathbf{z})) - \epsilon]$$

Figure 12: Latent traversal plots from β-VAE that learnt disentangled representations on the CelebA dataset.