Lecture 8: Autoencoders

Learning representation – no supervision

- How do we know what is relevant and what is not?

- **Main idea of unsupervised representation learning**: a new representation should be "smaller" but allow to recover the original one.

- *Why smaller?* (because we need to eliminate irrelevant, because we need something small)

Recap: PCA

We now want to pick the *d* optimal components $\{v_1, \dots v_D\} \in \mathbb{R}^N$

Let $\tilde{X} = [\tilde{x}_1, \tilde{x}_2 \dots \tilde{x}_M]$

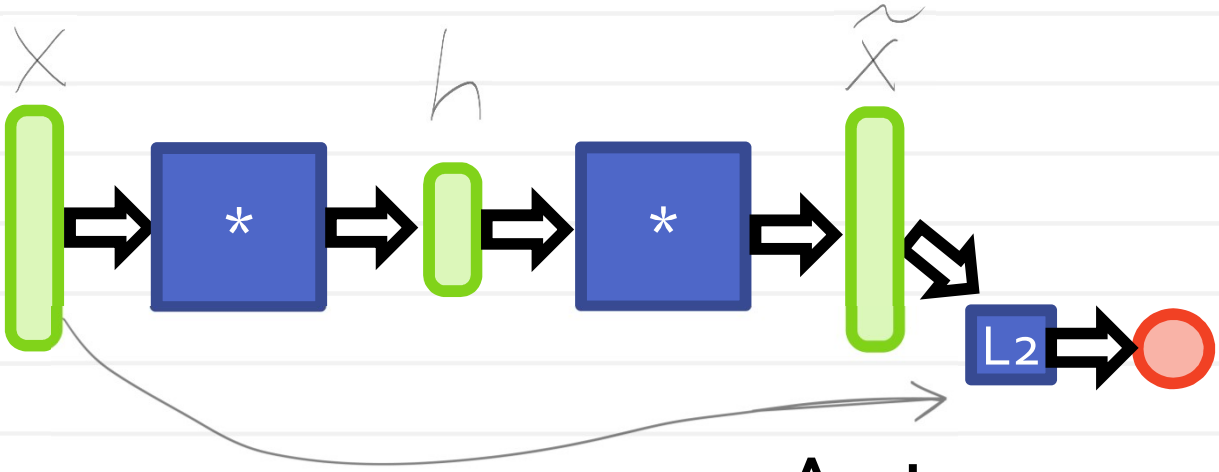
be the reconstruction of the original vectors.

We want to minimize the reconstruction error:

$$\sum_{i=1}^{M} \| x_i - \tilde{x}_i \|_2^2 \to \min$$

$$\| X - \tilde{X} \|_F^2 \to \min$$

Neural network version (autoencoder)



PCA:

$$h = U^T x$$

$$\tilde{x} = U h$$

$$\min_U \| x - \tilde{x} \|_2^2$$

$$s.t.: \langle u^i, u^j \rangle = \delta_{ij}$$

Autoencoder:

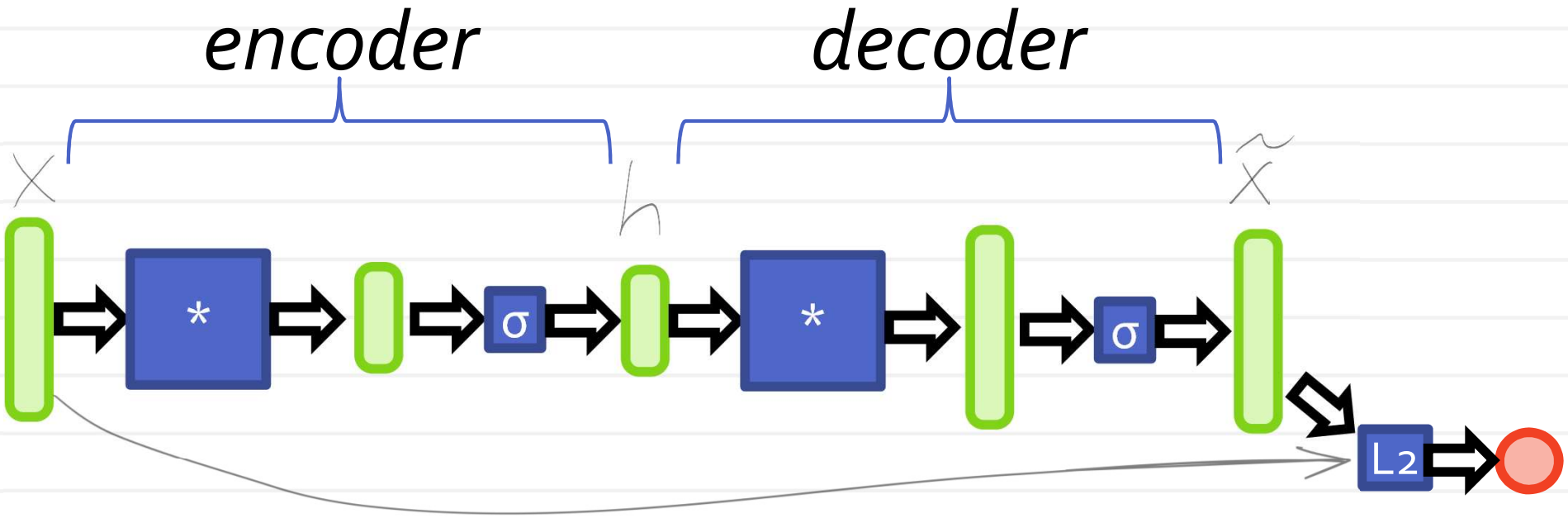
$$h = W x$$

$$\tilde{x} = W^T h$$

$$\min_W \| x - \tilde{x} \|_2^2$$

[Bengio et al. 07]

Autoencoder: adding non-linearity

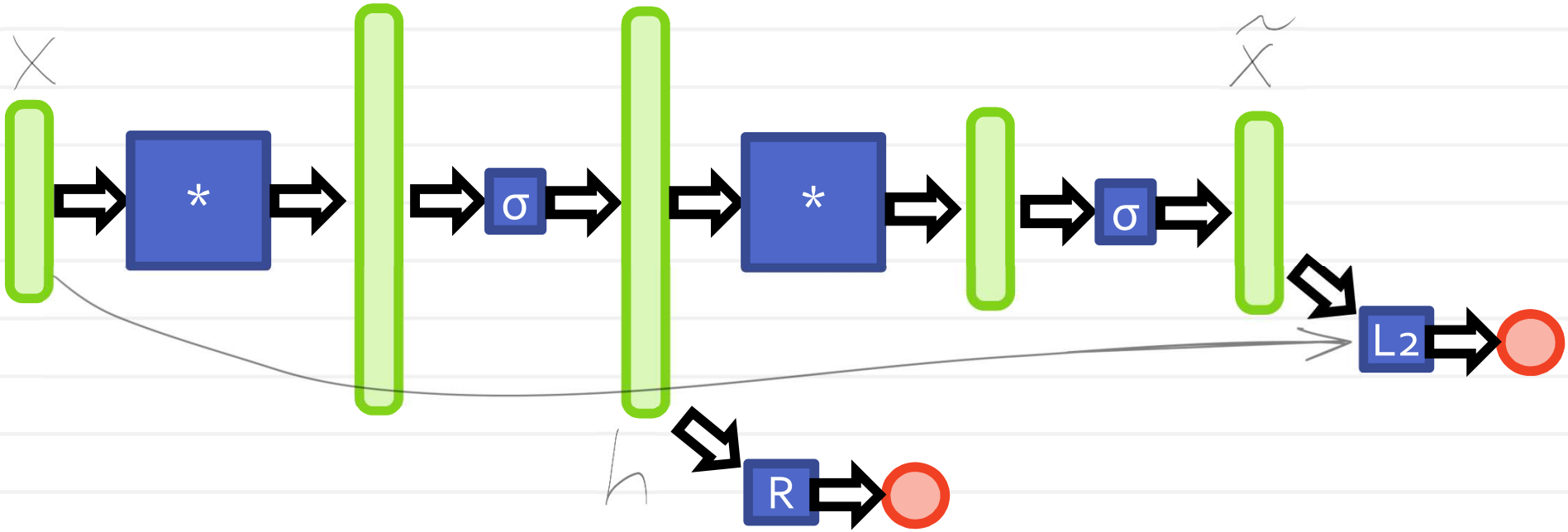


$$h = \sigma(Wx)$$

$$\tilde{x} = \sigma(W'h)$$

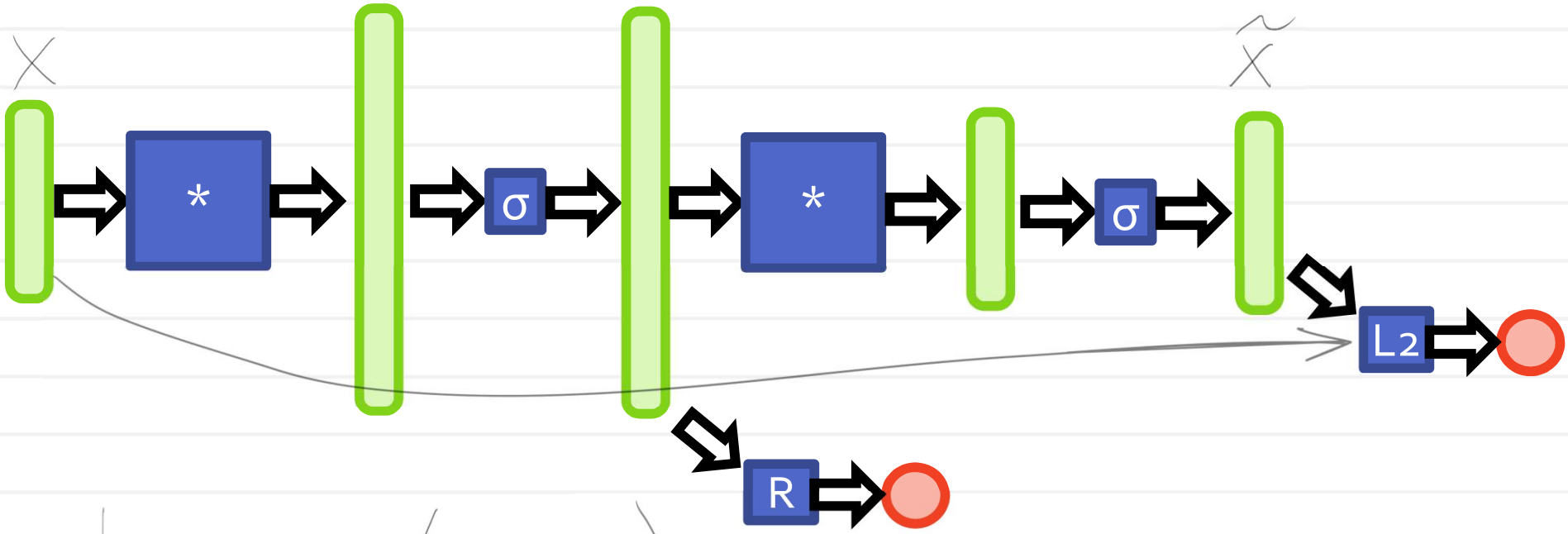
$$\min_{W, W'} \| x - \tilde{x} \|_2^2$$

Learning bigger/richer representation



- Naïve approach will learn an identity function
- Solution 1: regularize hidden representation

Sparse autoencoders



$$h = \sigma(Wx)$$

$$\tilde{x} = \sigma(W'h)$$

$$\sum |h_i| = \|\cdot\|_1$$

$$\min_{W,W'} \|x - \tilde{x}\|_2^2 + \lambda R(h)$$

Sparse autoencoders

$$\min_{W, W'} \| X - \tilde{X} \|_2^2 + \lambda R(h)$$

estimating from e.g. minibatch

$$R(h) = \sum_i KL(\text{Binomial}(\rho_i) \| \text{Binomial}(\hat{\rho}_i))$$

$$= \sum_i \rho_i \log \frac{\rho_i}{\hat{\rho}_i} + (1-\rho_i) \log \frac{1-\rho_i}{1-\hat{\rho}_i}$$

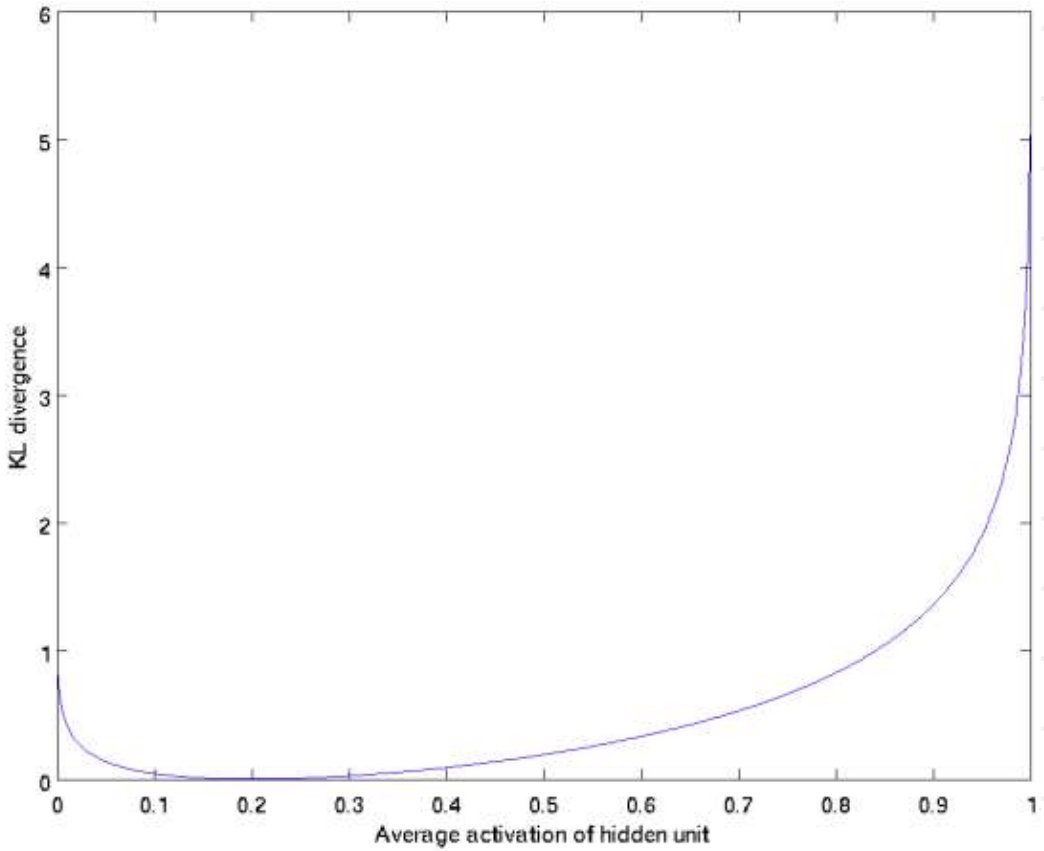
$$\frac{\partial R}{\partial \hat{\rho}_i} = -\frac{\rho_i}{\hat{\rho}_i} + \frac{1-\rho_i}{1-\hat{\rho}_i}$$

[Ng 2011]

Sparse autoencoders

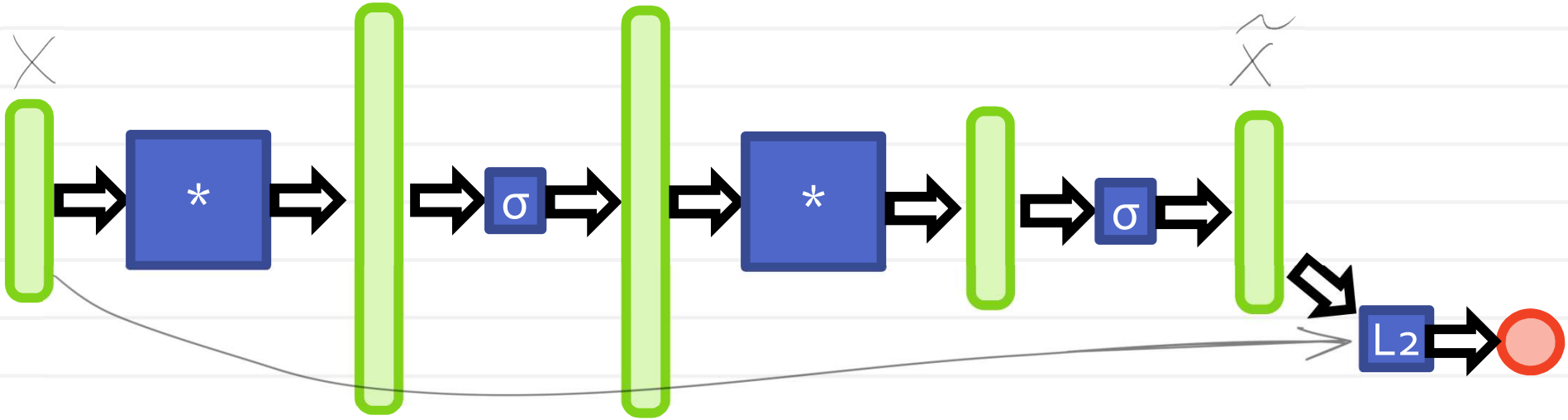
$$R(h) = \sum_i KL\left(\text{Binomial}(\rho_i) \,\|\, \text{Binomial}(\hat{\rho}_i)\right)$$

$$= \sum_i \rho_i \log \frac{\rho_i}{\hat{\rho}_i} + (1-\rho_i) \log \frac{1-\rho_i}{1-\hat{\rho}_i}$$



[Ng 2011]

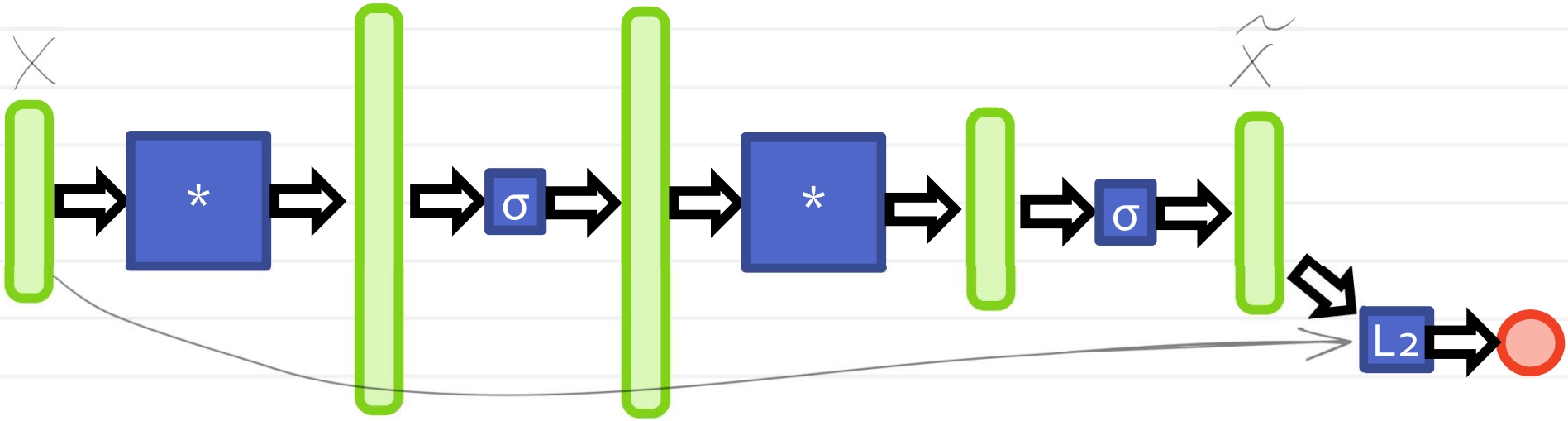
Learning bigger representation



- Naïve approach will learn an identity function
- Solution 2: corrupt the input and reconstruct uncorrupted

[Vincent et al. 2010]

Denoising autoencoder



$$h = g(W(x + n))$$

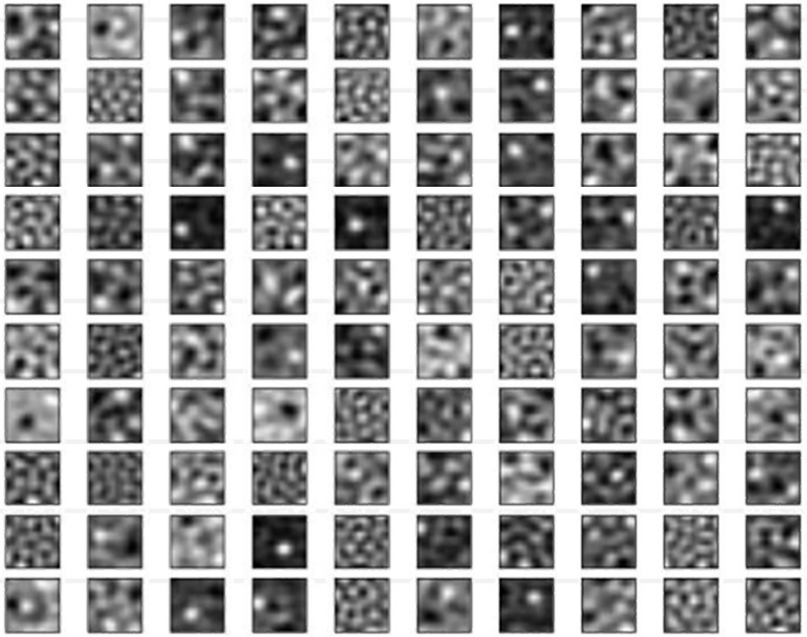
$$\tilde{x} = g(W'h)$$

$$\min_{W,W'} \sum_i \| x_i - \tilde{x}_i \|_2^2$$

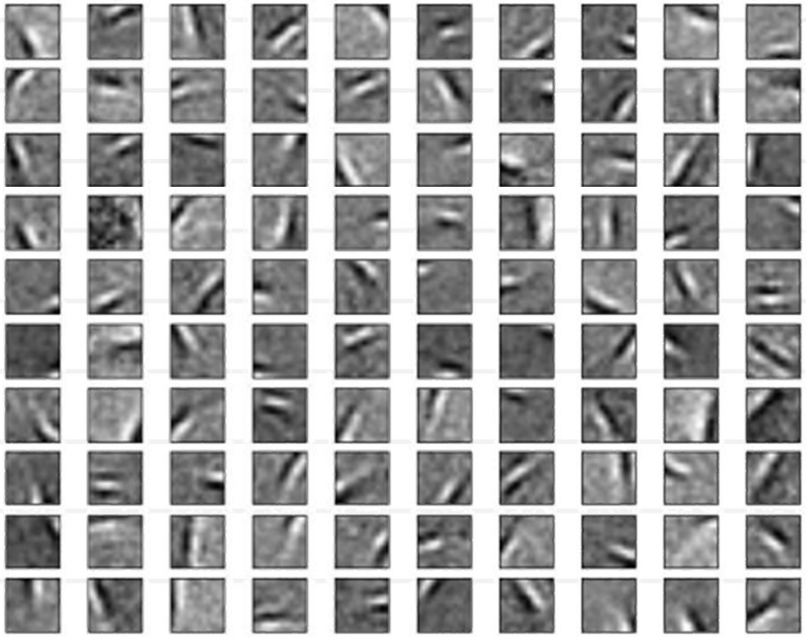
[Vincent et al. 2010]

Denoising autoencoder

12x12 patches, 200 hidden units

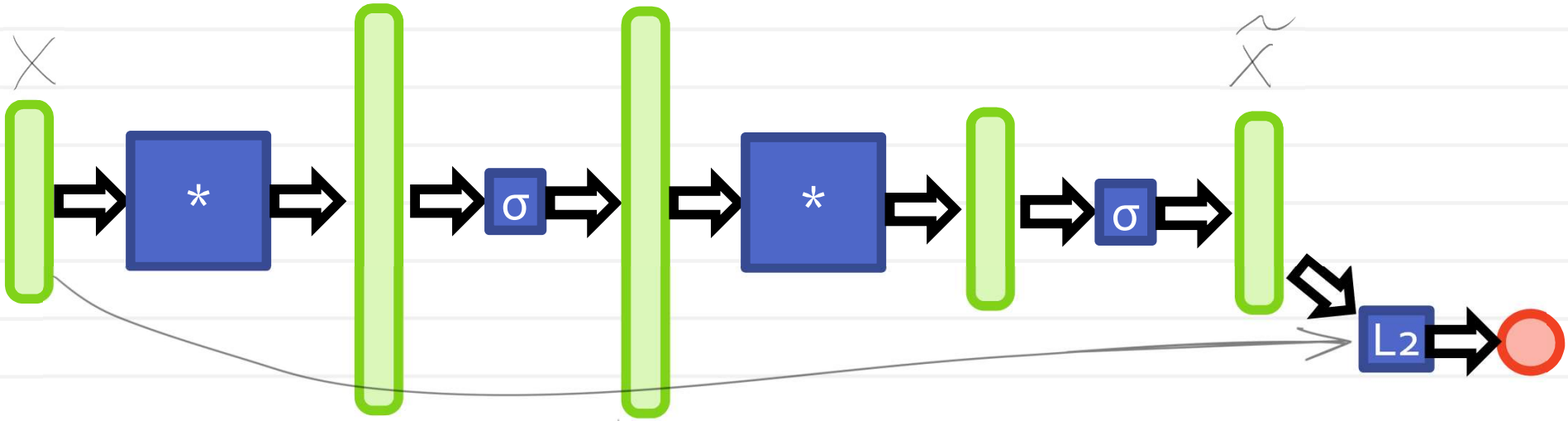


Autoencoder with weight-decay



Denoising autoencoder (Gaussian noise) [Vincent et al. 2010]

Contractive autoencoders



$$h = \sigma(Wx)$$

$$\tilde{x} = \sigma(W'h)$$

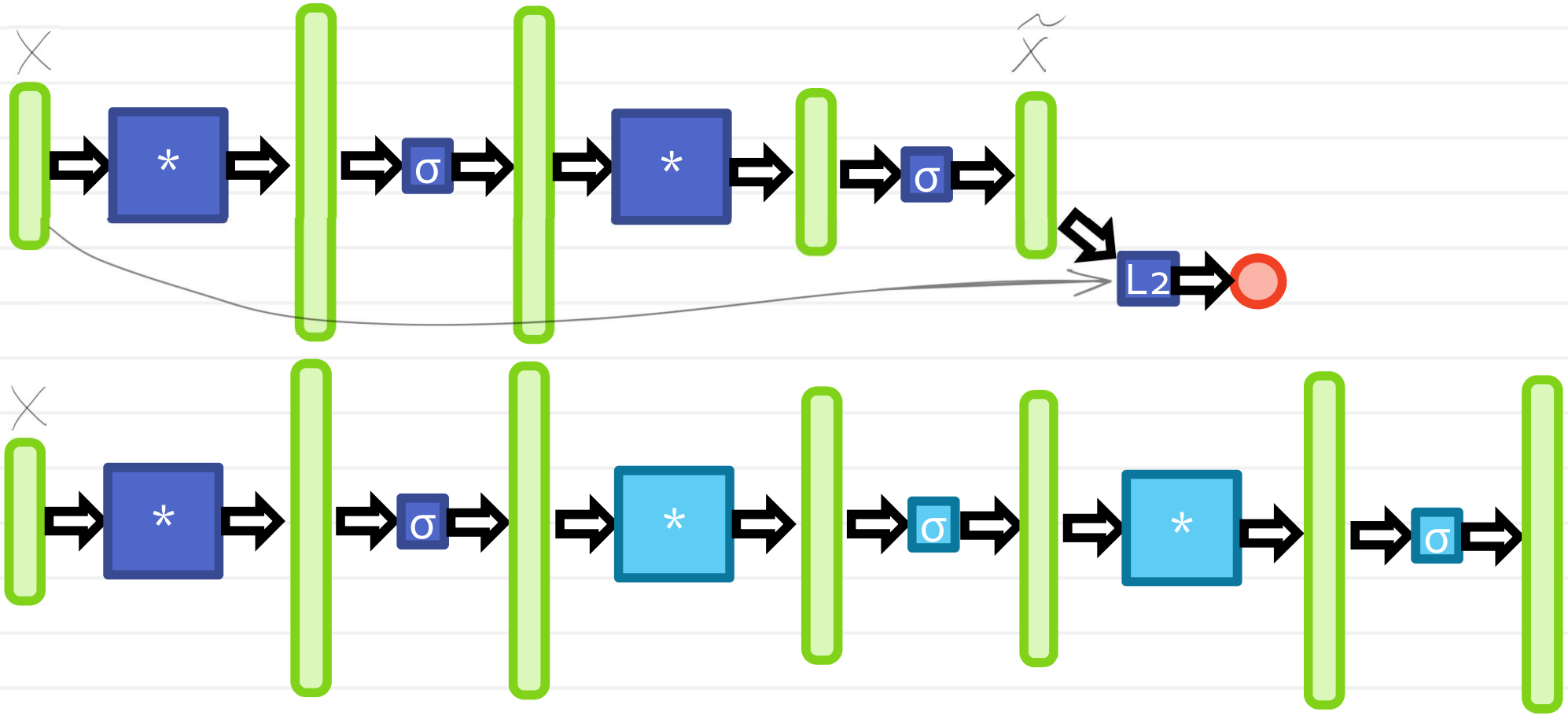
encoder

$$\min_{W,W'} \|x - \tilde{x}\|_2^2 + \lambda \left\| \frac{\partial h_i(x)}{\partial x_j} \right\|_F^2$$

$$\frac{\partial h_i}{\partial x_j} = (h_i(1-h_i))\, W_{ij}$$

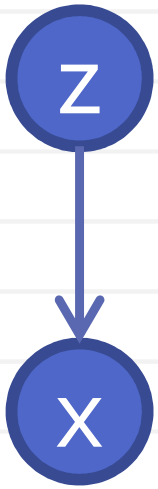
[Rifai et al. 2011]

Training deep autoencoders



- Historically: layerwise pretraining
- Most modern implementations: end-to-end
- Commonly including convolutional and up-convolutional layers

Maximum likelihood



$p_\theta(x \mid z)$

Very natural idea: train θ by maximum likelihood:

$$\log p_\theta(x) = \log \int_z p_\theta(x, z)\, dz \to \max$$

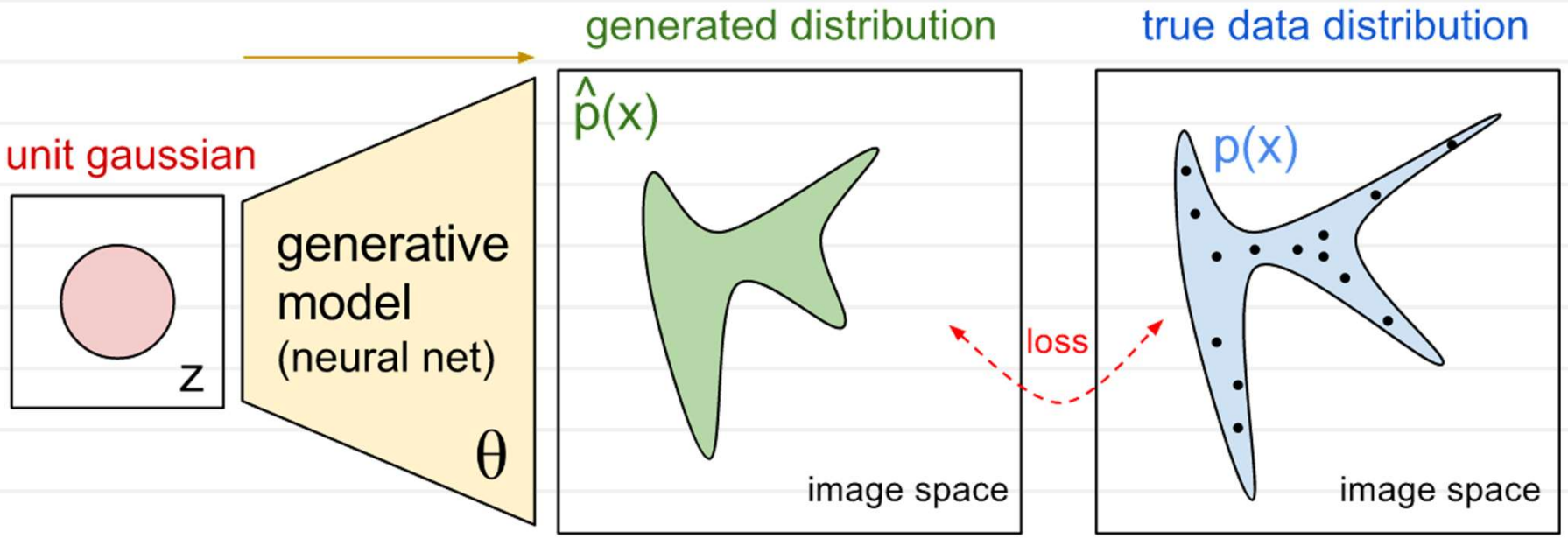
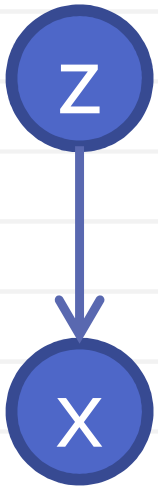


Image credit: OpenAI blog

Maximum likelihood via VAE



$p_\theta(x|z)$

Natural idea: train θ by maximum likelihood.

$$\log p_\theta(x) = \log \int_z p_\theta(x,z)\,dz \to \max$$

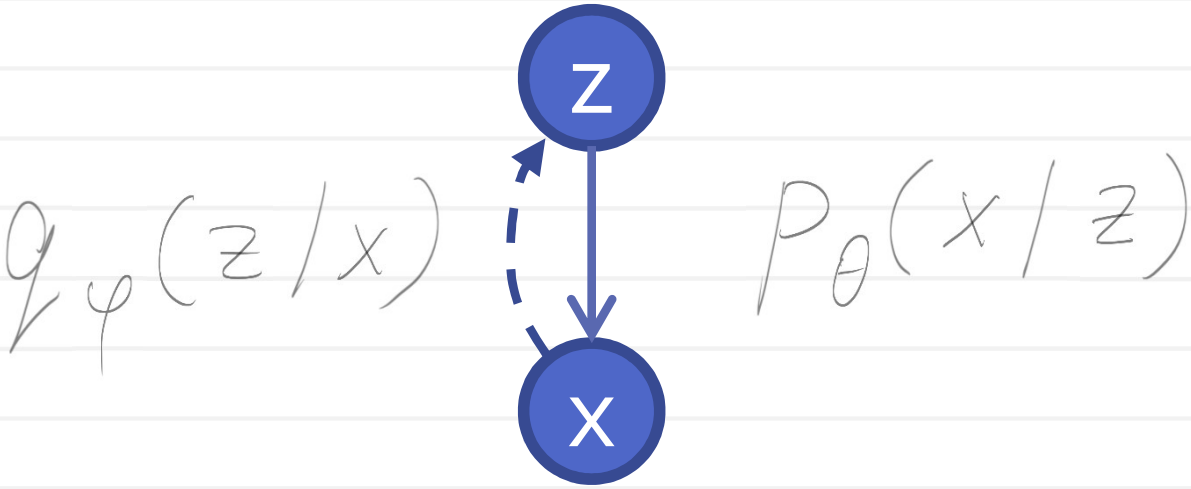
The integral is intractable.

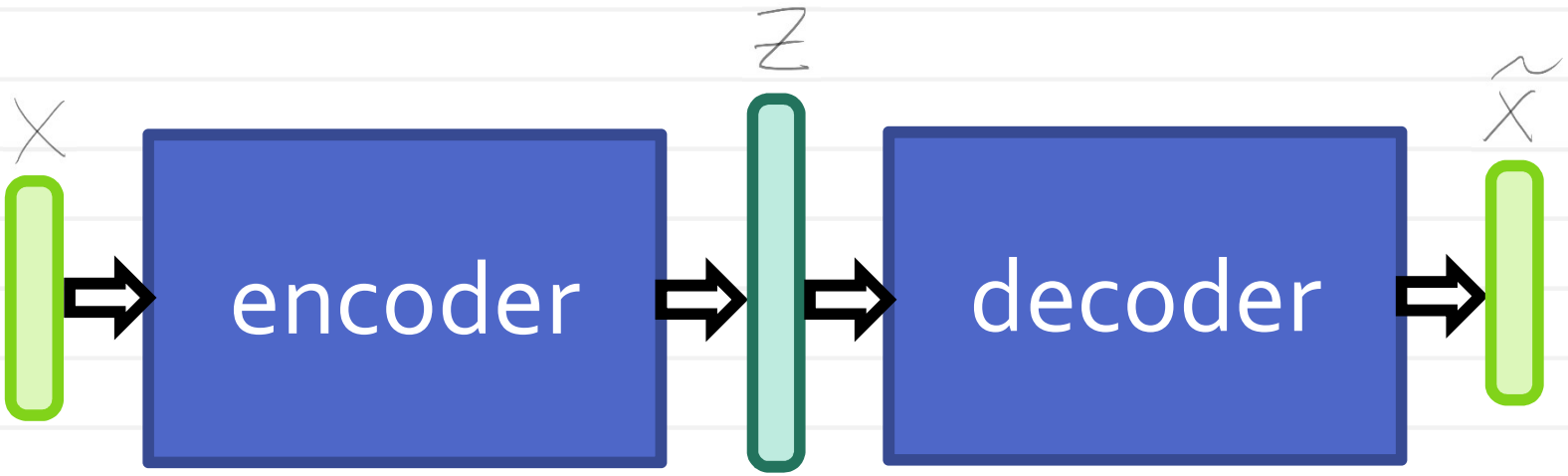
VAE solution: introduce an *encoder*

$$q_\varphi(z|x) \approx p(z|x)$$

Derive an approximation to ML objective

Variational autoencoder as a graphical model





[Kingma & Welling 14]

Variational lower bound

$$\log p(x) = \log \int_z p(x,z)\,dz =$$

$$= \log \int_z p(x,z)\frac{q(z|x)}{q(z|x)}\,dz =$$

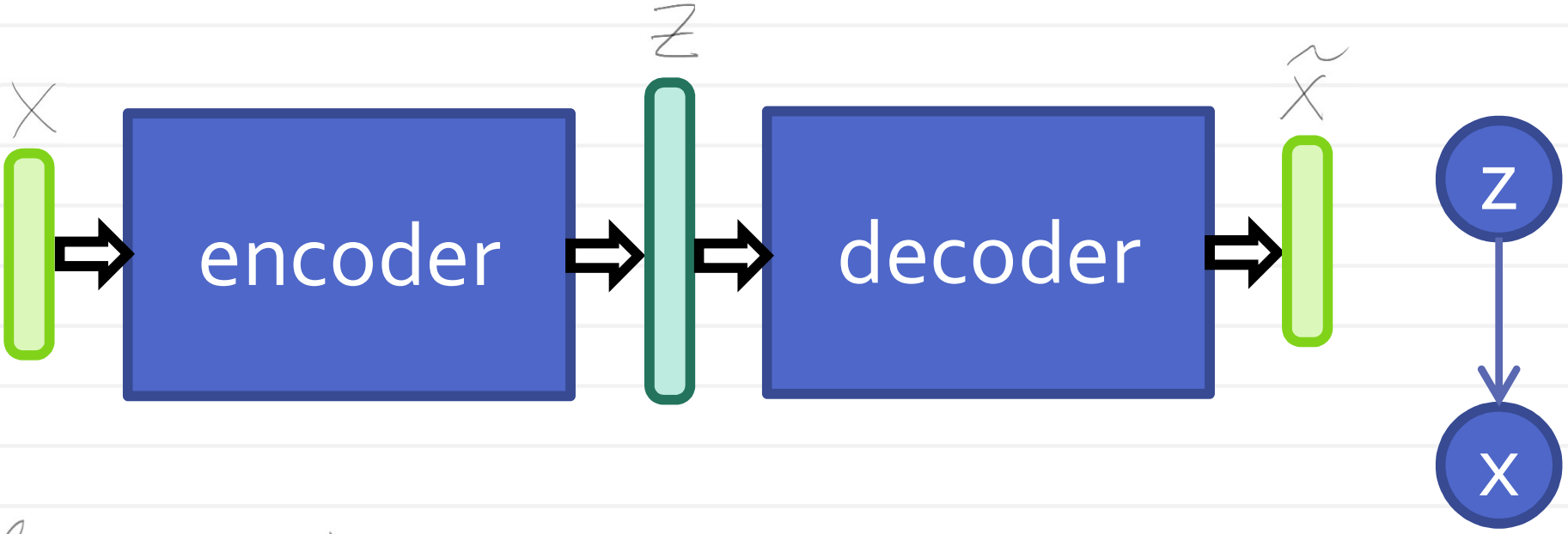
$$= \log E_{q(z|x)} \frac{p(x,z)}{q(z|x)} =$$

$$\log E_{q(z|x)} \frac{p(x|z)\,p(z)}{q(z|x)} \geq$$

$$\geq E_{q(z|x)} \log p(x|z) + E_{q(z|x)} \log \frac{p(z)}{q(z|x)} =$$

$$= E_{q(z|x)} \log p(x|z) - KL(q(z|x)\,\|\,p(z))$$

Variational lower-bound

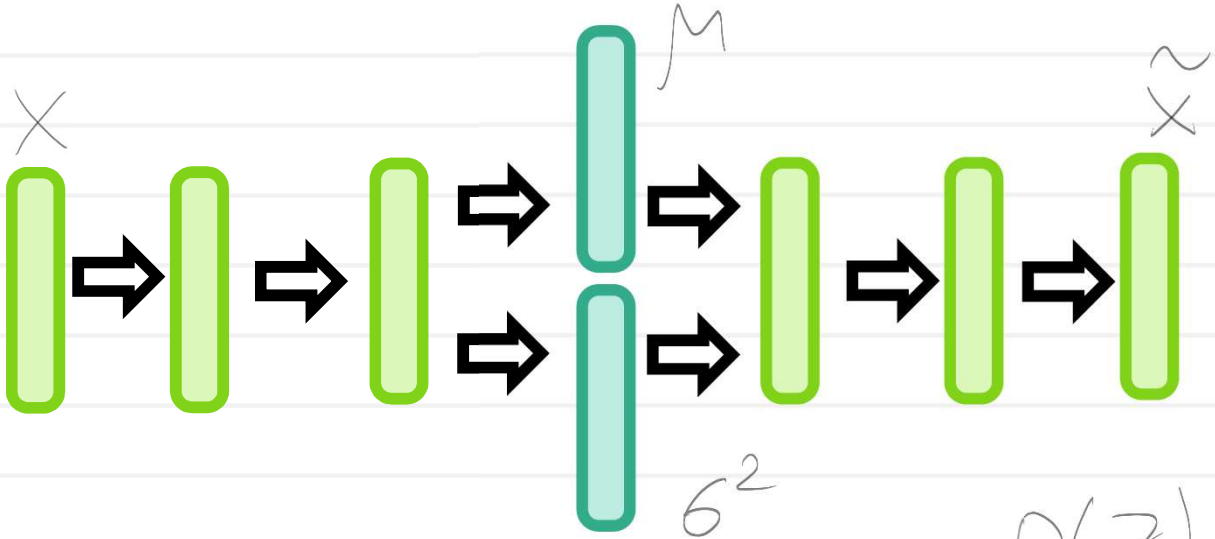


$$\log p(x) \geq$$

$$\underbrace{-KL(q_\varphi(z|x) \| p(z))}_{\text{regularization}} + \underbrace{E_{q_\varphi(z|x)}[\log p_\theta(x|z)]}_{\sim\text{“denoising auto-encoder”}}$$

[Kingma & Welling 14]

Minimizing regularization term



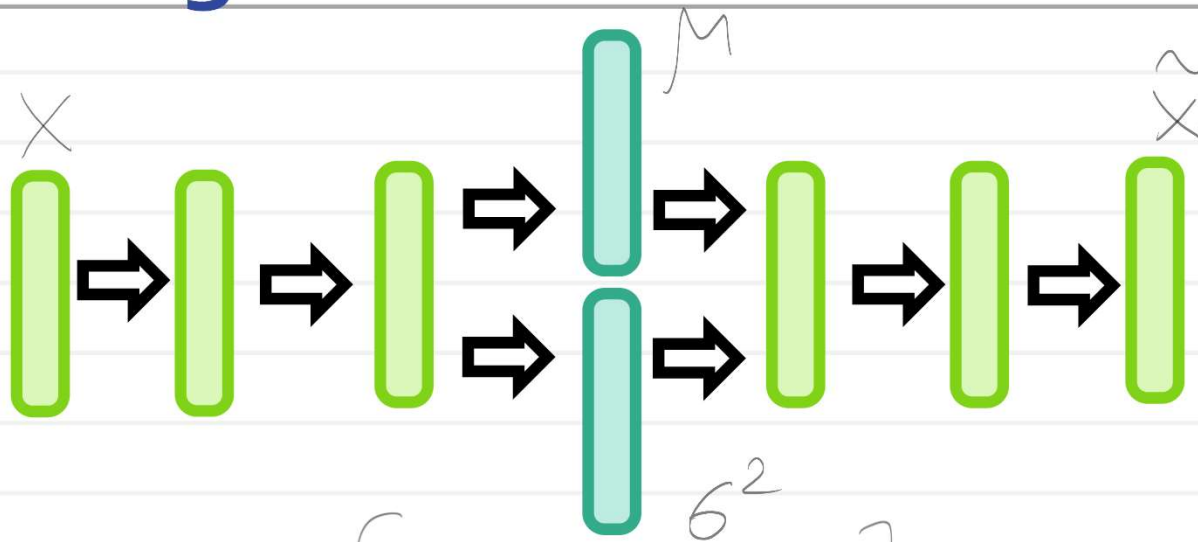
$$p(z) = \prod \mathcal{N}(0,1)$$

$$KL(q_\varphi(z/x) \,||\, p(z)) =$$

$$= \frac{1}{2} \sum \left(\mu_j^2 + \sigma_j^2 - 1 - \log \sigma_j^2 \right)$$

[Kingma & Welling 14]

Optimizing reconstruction term



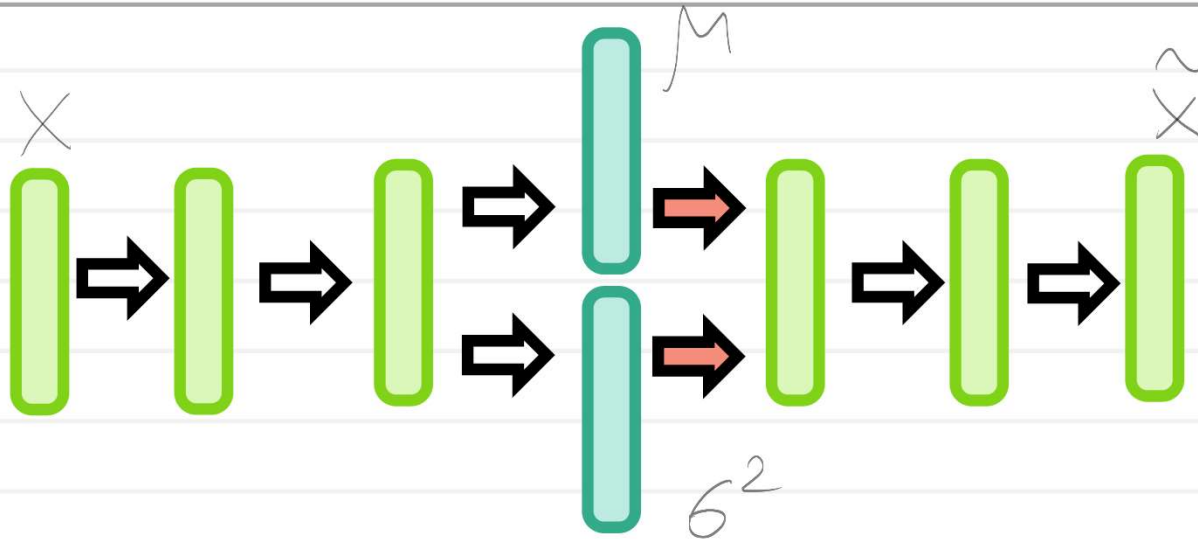
$$E_{q_\varphi(z|x)}\left[\log p_\theta(x|z)\right] \longrightarrow \max$$

$$p_\theta(x|z) \propto \exp\left(-\|x-\tilde{x}\|^2\right)$$

$$E_{q_\varphi(z|x)}\|d_\theta(z) - x\|_2^2 \longrightarrow \min$$

[Kingma & Welling 14]

Reparameterization trick



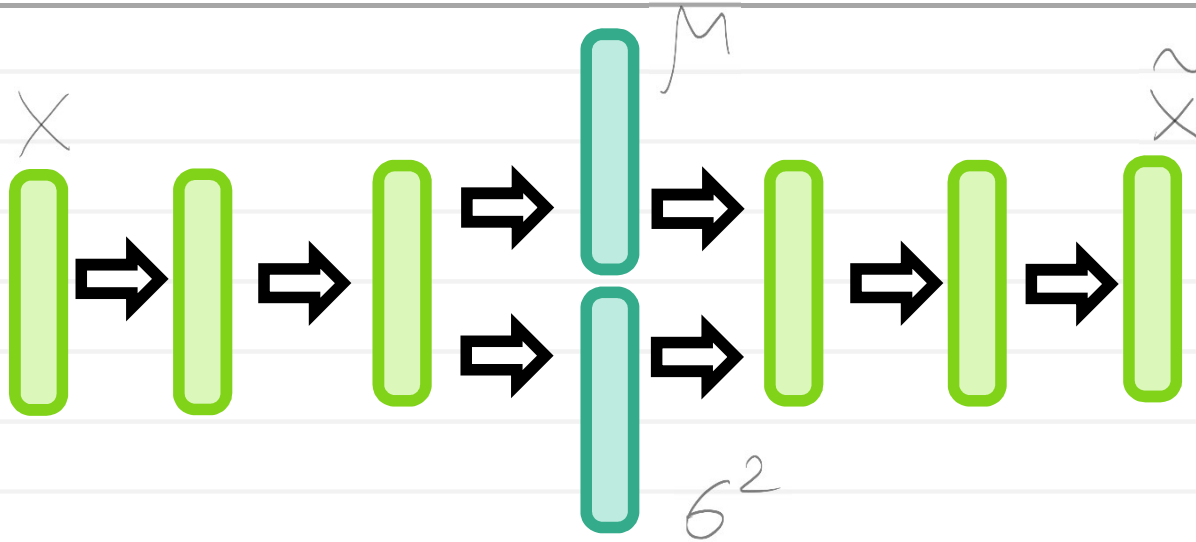
$$E_{q_\varphi(z|x)} \| d_\theta(z) - x \|_2^2 \to \min$$

$$\tilde{z} = \mu + \sigma^2 \odot \varepsilon \sim \mathcal{N}(0; i)^N$$

$$L = \| d_\theta(\tilde{z}) - x \|_2^2 \to \min$$

[Kingma & Welling 14]

Reparameterization trick



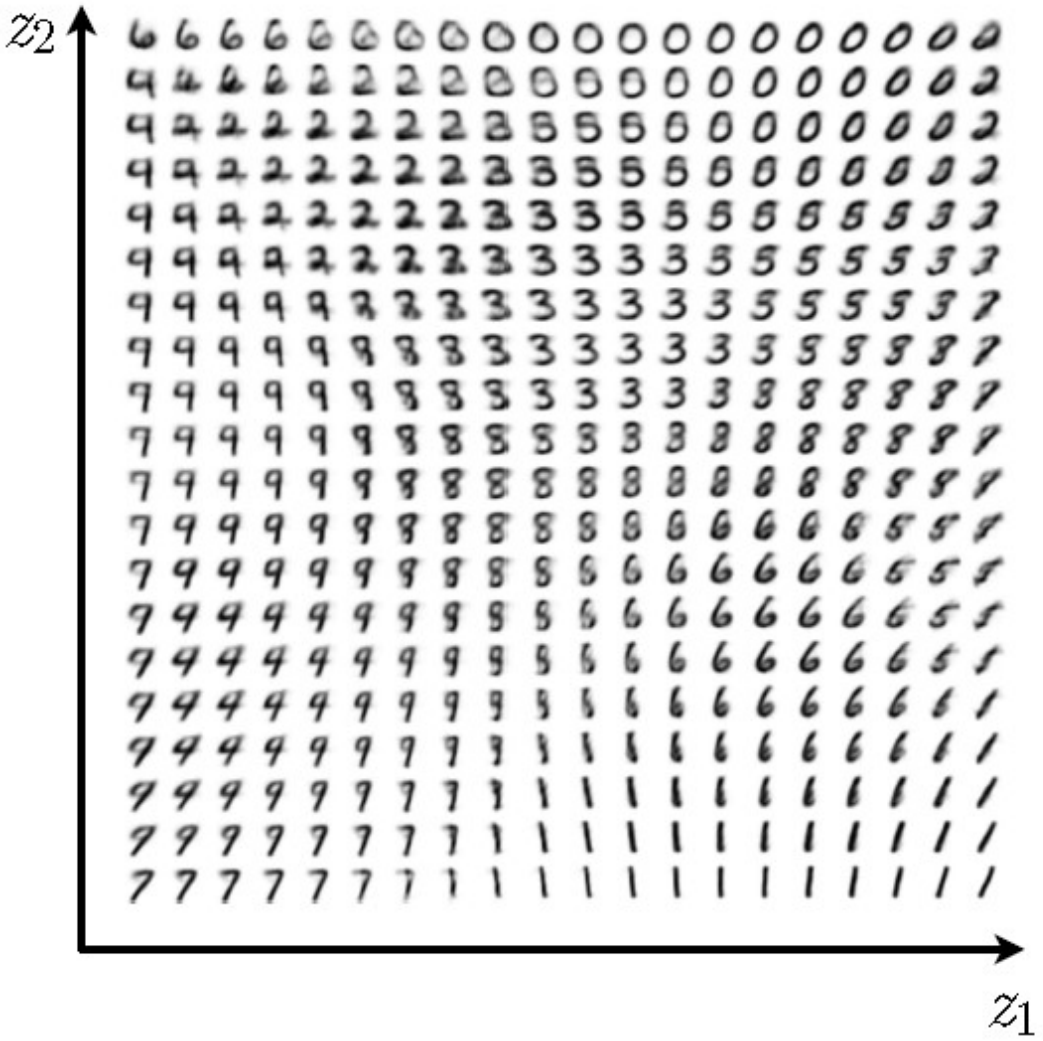
$$\tilde{z} = \mu + \sigma^2 \odot \varepsilon \sim \mathcal{N}(0; I)^N$$

$$\frac{dL}{d\mu} = \frac{dL}{d\tilde{z}} \qquad \frac{dL}{d\sigma^2} = \frac{dL}{d\tilde{z}} \odot \varepsilon$$

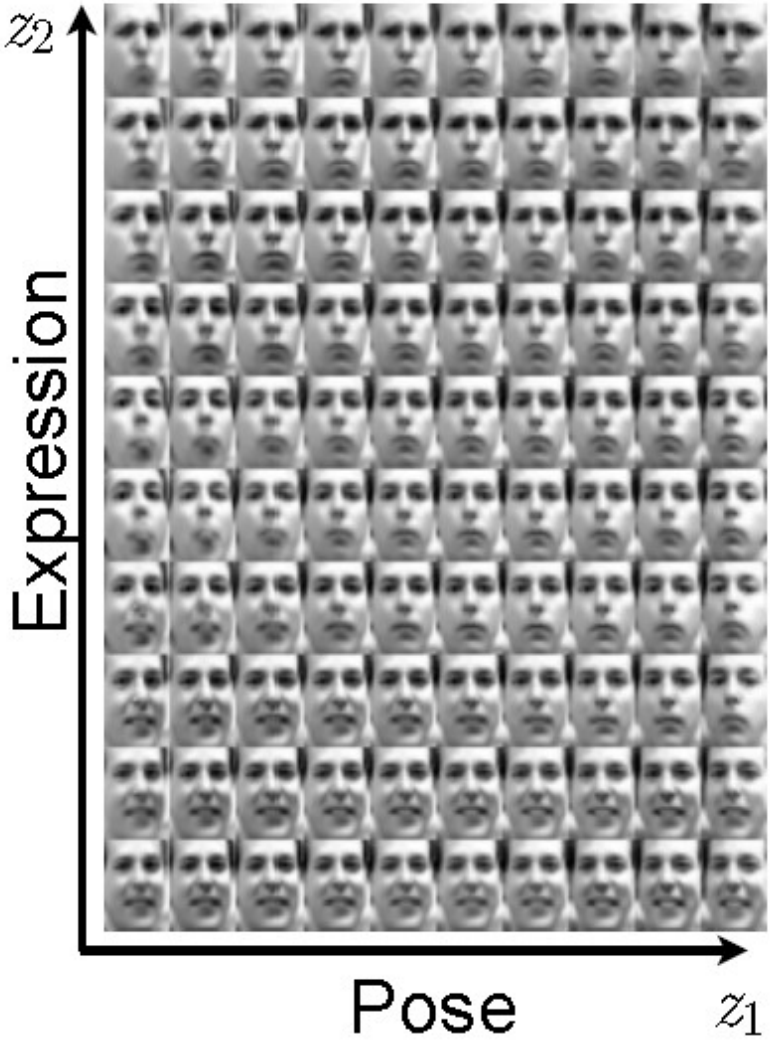
[Kingma & Welling 14]

VAE-learned manifolds

MNIST:



Frey Face dataset:



[Kingma & Welling 14]

VAE-learned manifolds

(a) 2-D latent space

(b) 5-D latent space

(c) 10-D latent space

(d) 20-D latent space

[Kingma & Welling 14]

Collapsing components

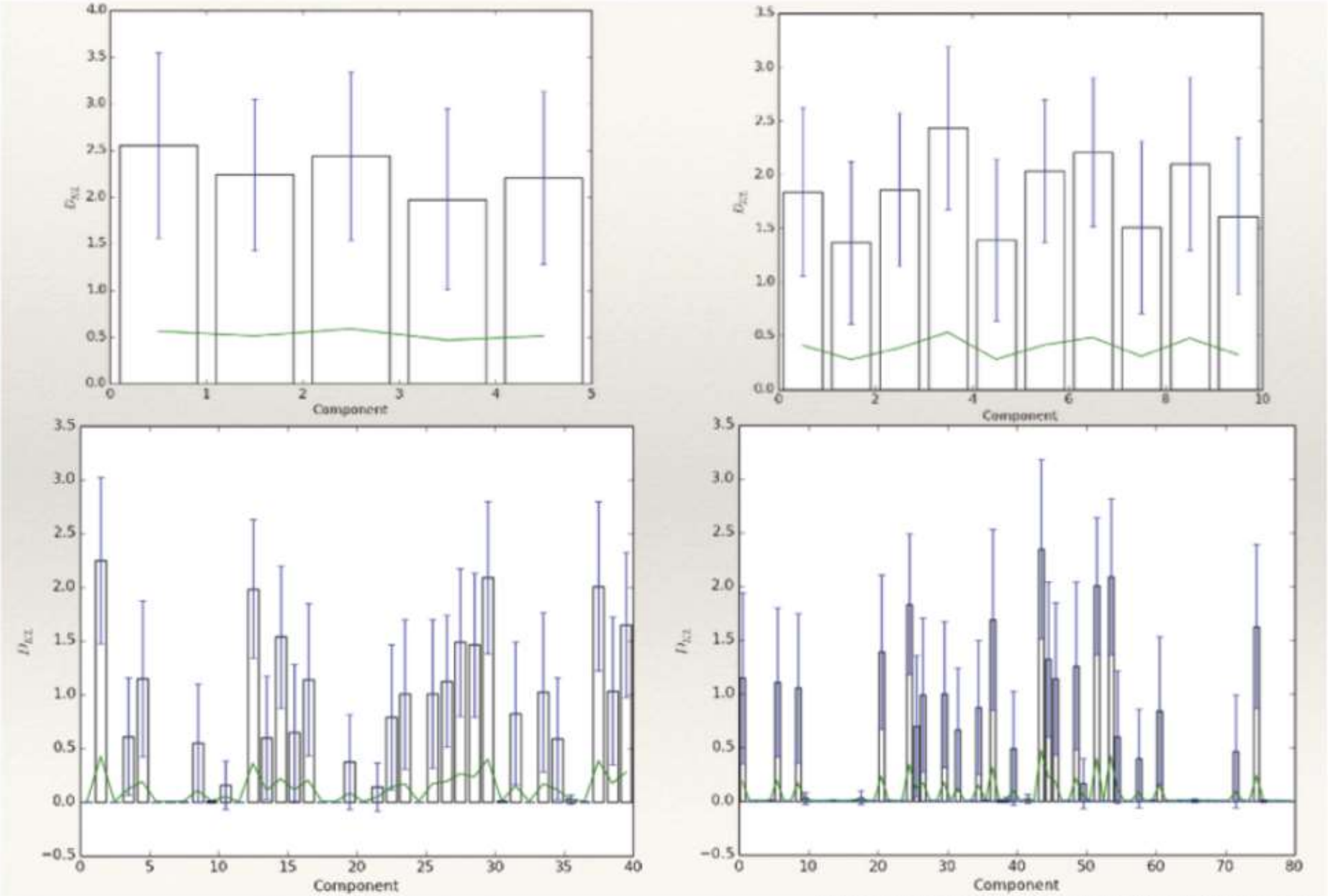


Figure from Laurent Dinh & Vincent Dumoulin

Collapsing components

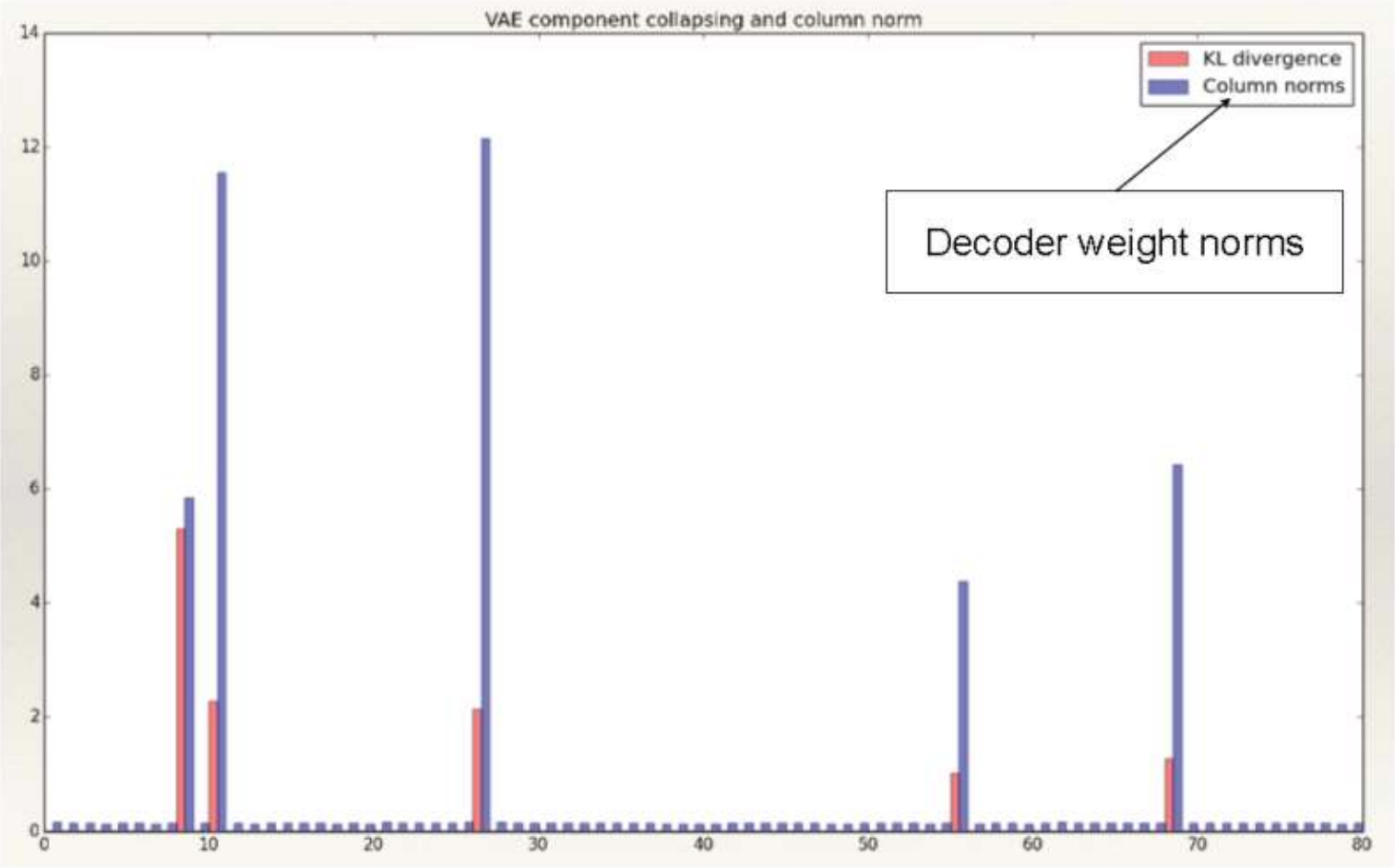


Figure from Laurent Dinh & Vincent Dumoulin

- Components that are shrunk to *N(0,1)* are not used by decoder

Sample quality comparison

VAE

GAN

Source: OpenAI blog

Sample quality comparison

VAE:

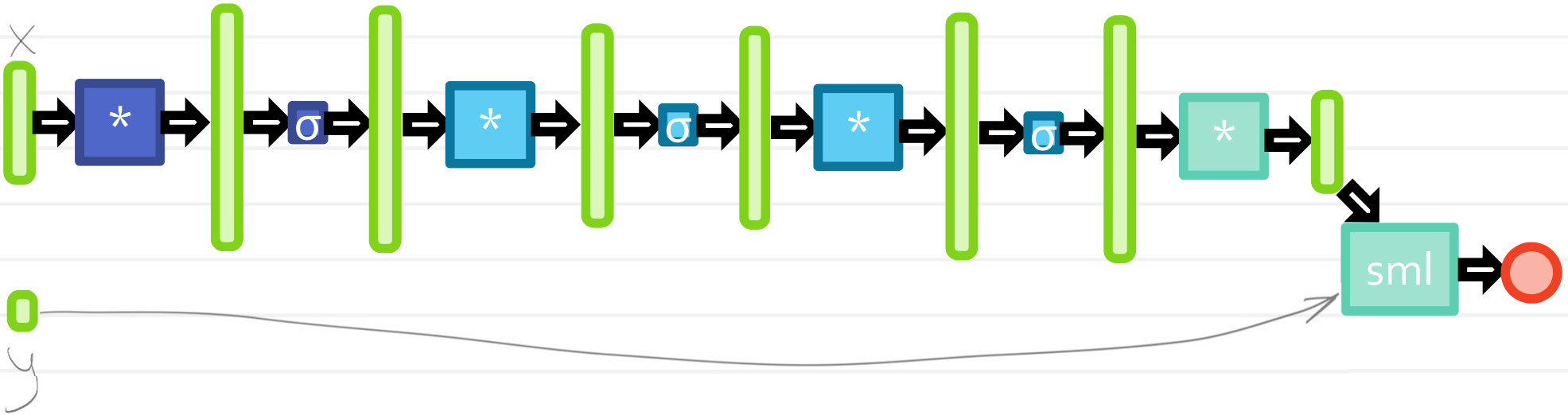
Source:
OpenAI blog

Sample quality comparison

DCGAN:

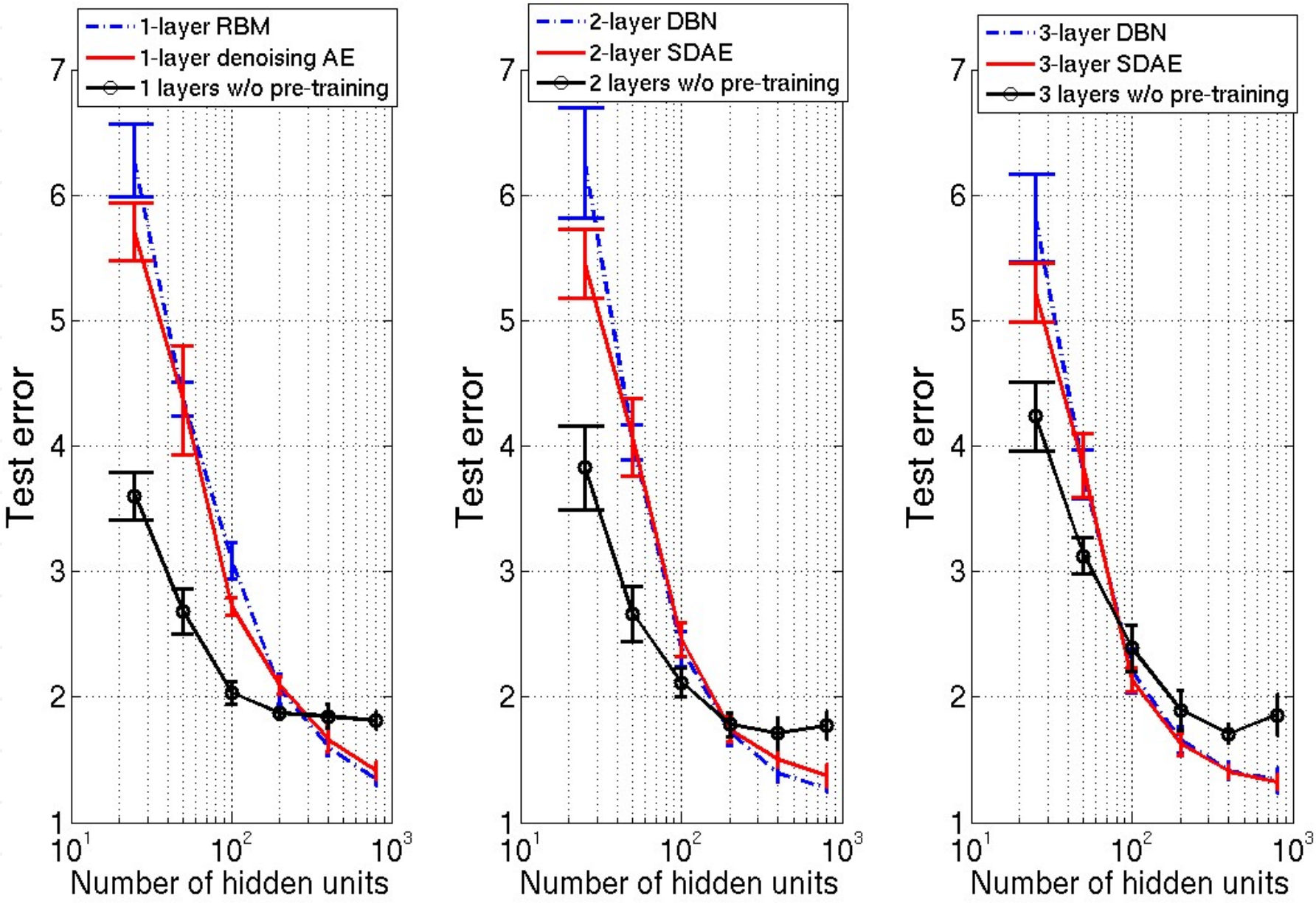
Source:
OpenAI blog

Using auto-encoders for pre-training



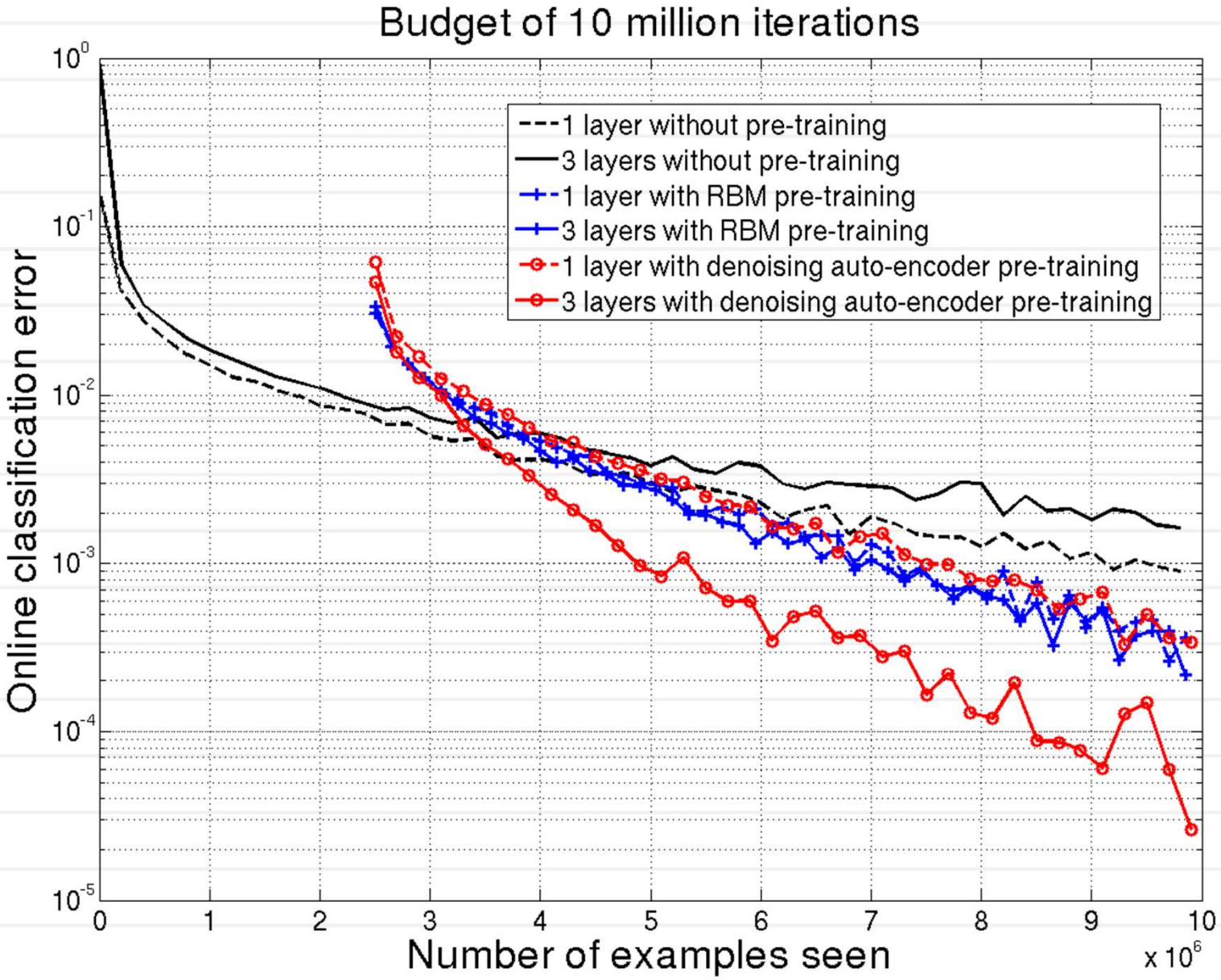
- Autoencoder is used to initialize a deep supervised feedforward architecture

Pretraining by unsupervised learning



[Erhan et al. 2010]

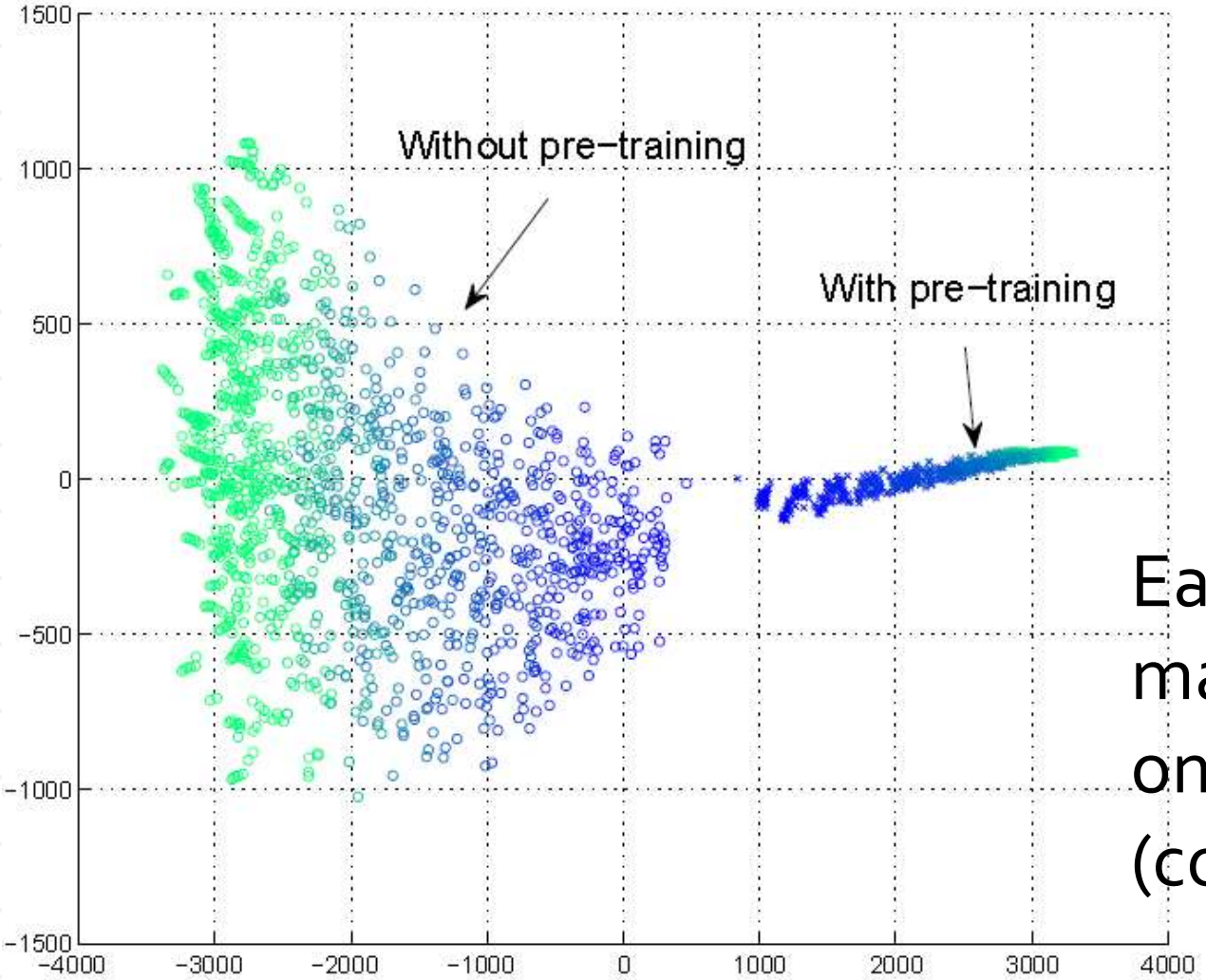
Pretraining by unsupervised learning



[Erhan et al. 2010]

Pretraining by unsupervised learning

Isomap of model space (greener = later)

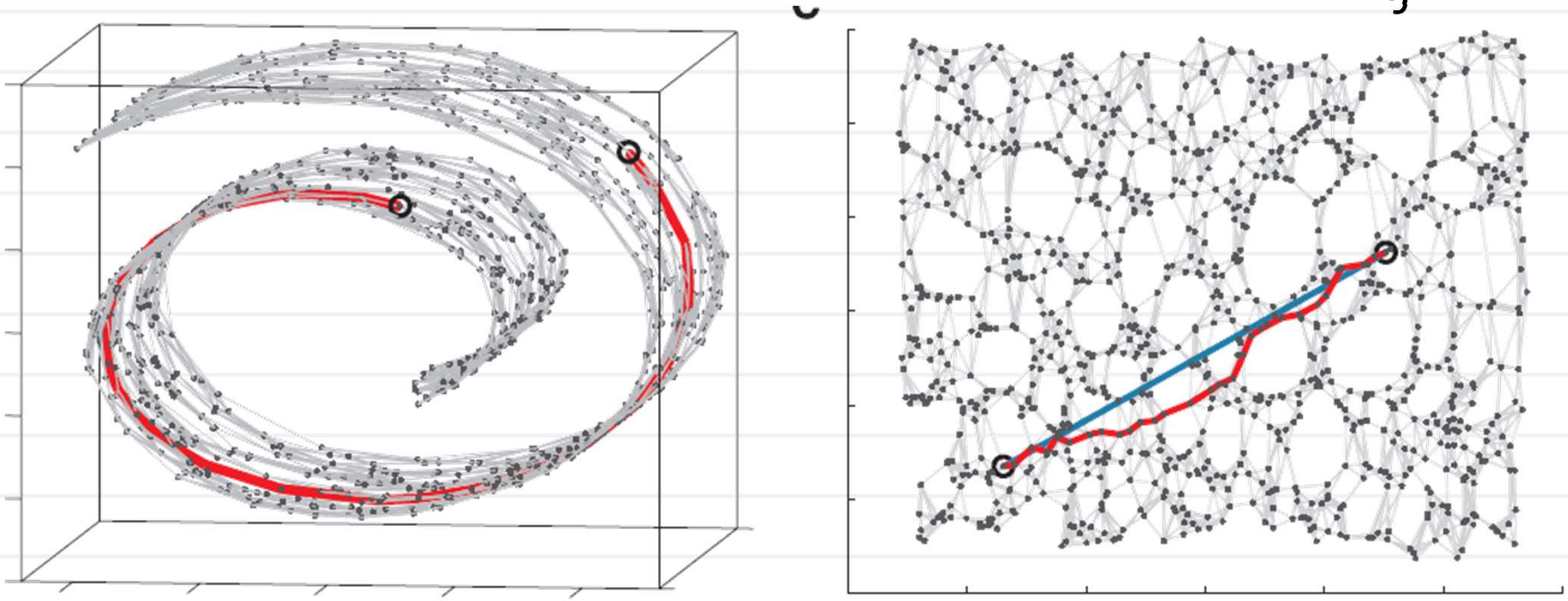


Each network is mapped to its output on a validation set (concat. posteriors)

[Erhan et al. 2010]

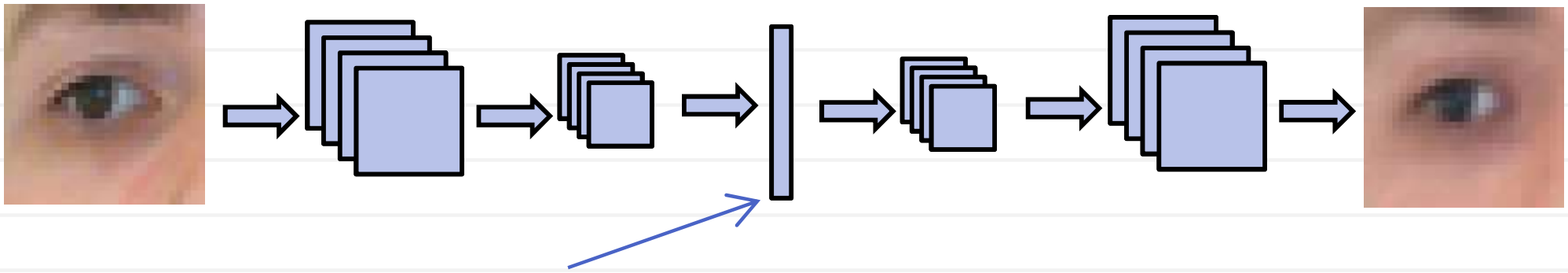
Isomap manifold learning

- Build a NN graph (unless your data are already a graph!)
- d_{ij} = shortest_path(V_i, V_j)
- Do MDS (project while preserving d_{ij})

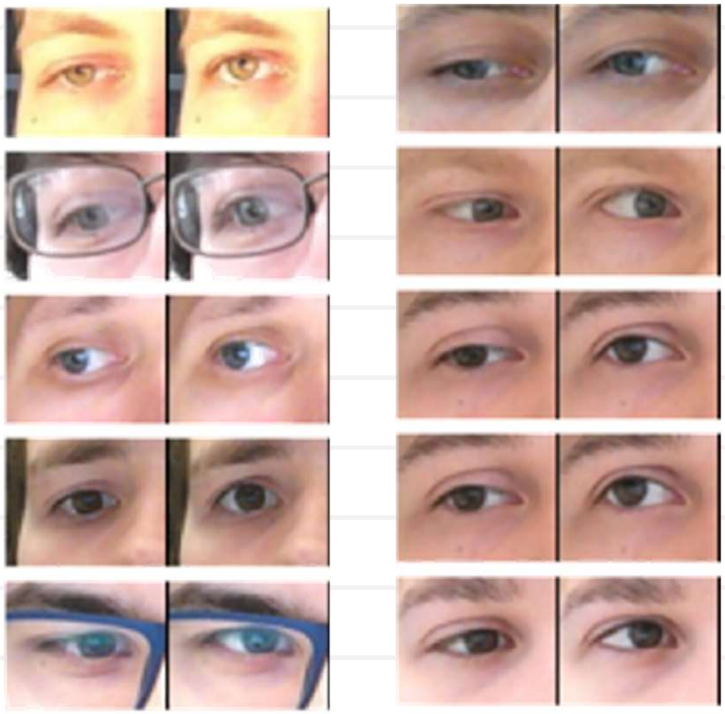


[Tenenbaum et al. *Science'2000*]

Smart image editing

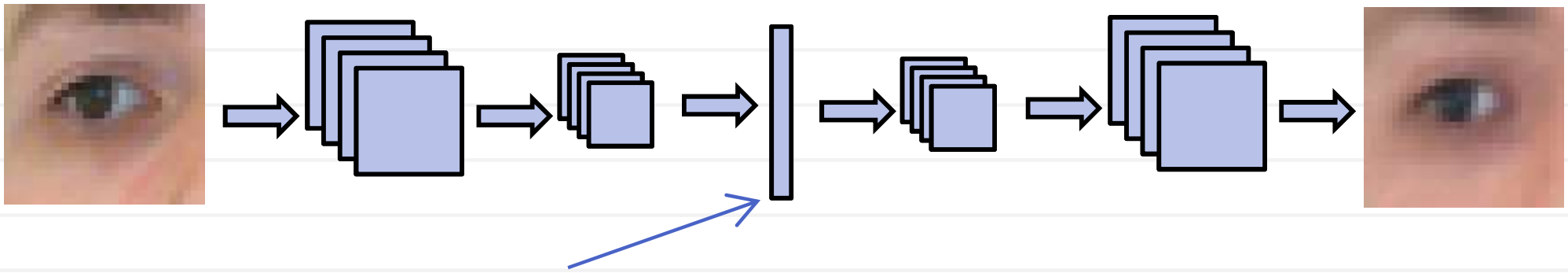


Low-dim space, where we can estimate semantically meaningful directions

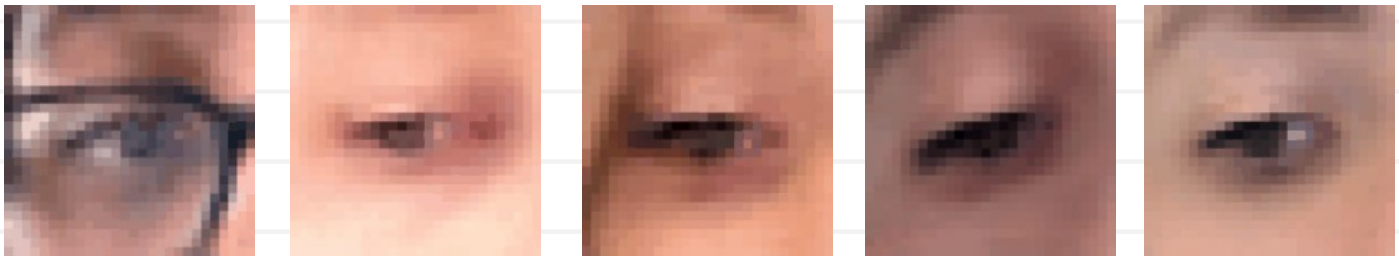


Given a bunch of pairs we can estimate a vector for gaze redirection

Latent space image editing



Low-dim space, where we can estimate semantically meaningful directions

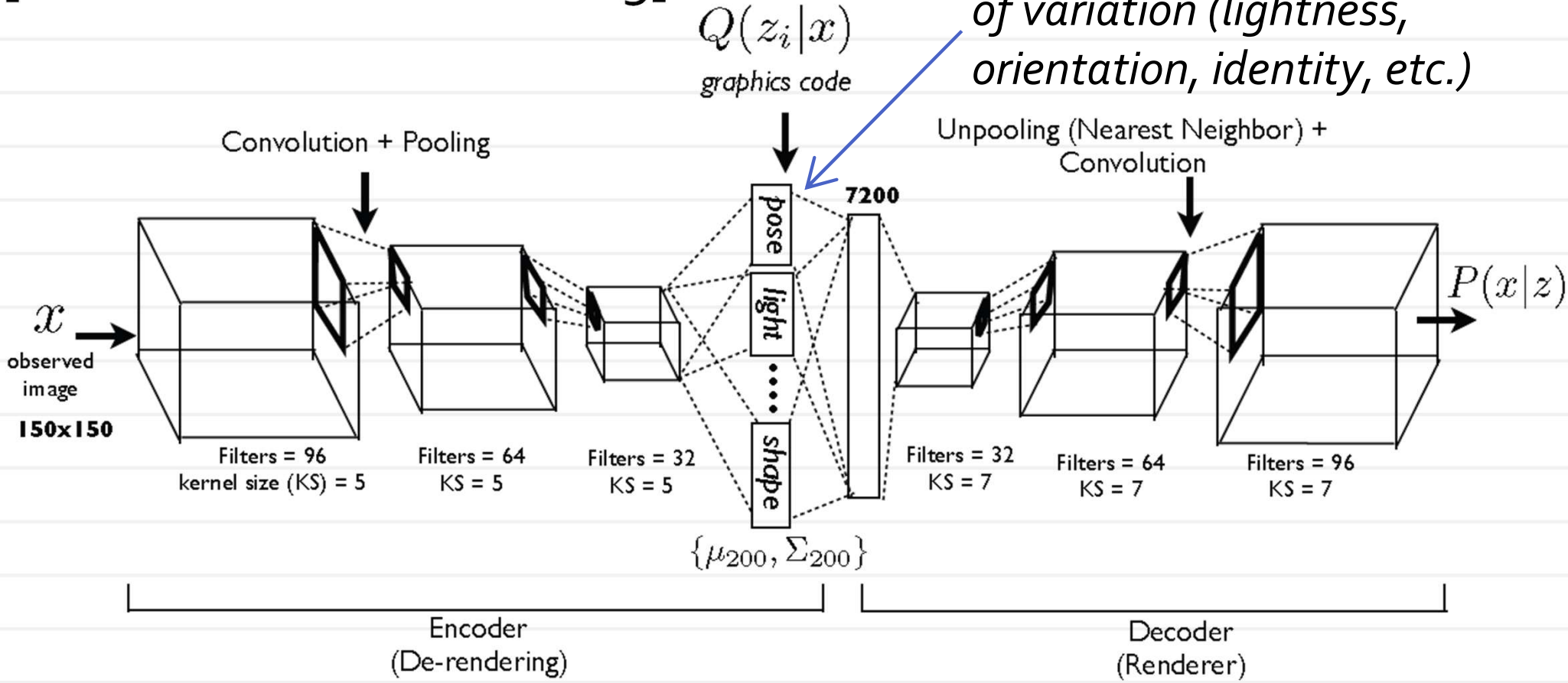


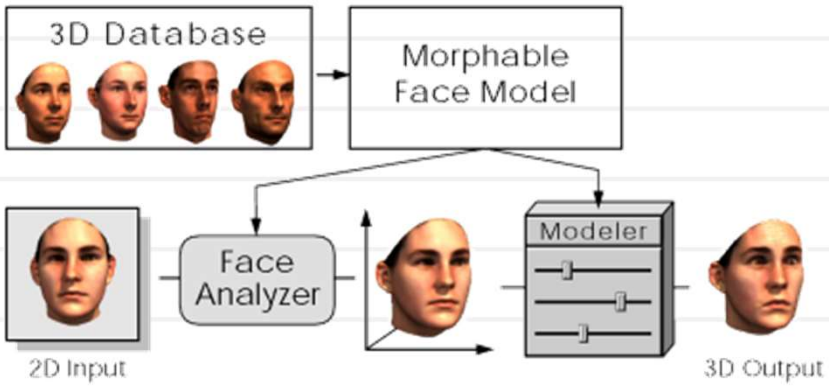
Thanks to L.Yekimov
(+F.Chervinsky,D.Kononenko,D.Sungatullina,Y.Ganin)

Deep Inverse Graphics

[Kulkarni et al. NIPS 2015]

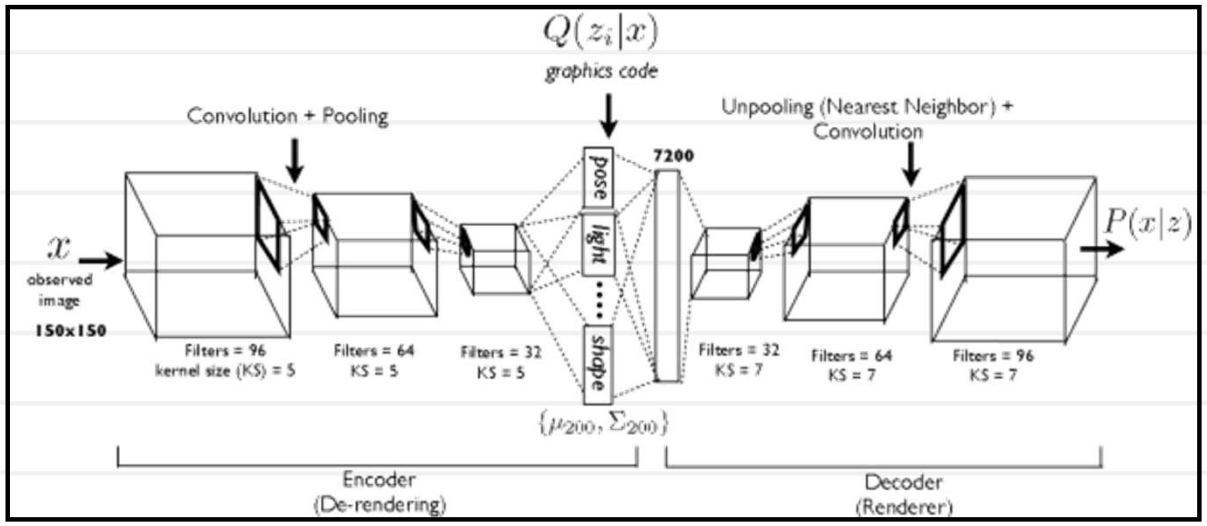
Different variables z are assigned to different factors of variation (lightness, orientation, identity, etc.)





[Blanz and Vetter 1999] morphable model

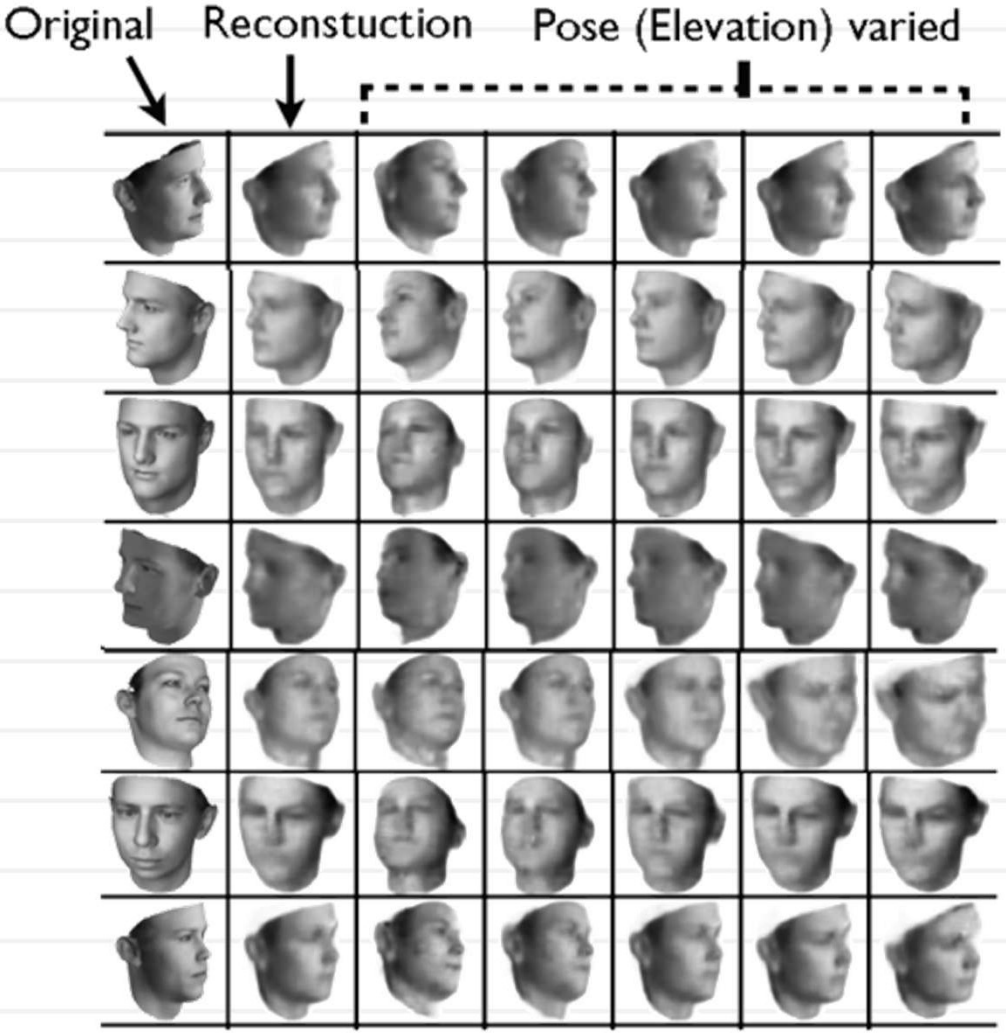
Ensuring semantic meaning of dimensions

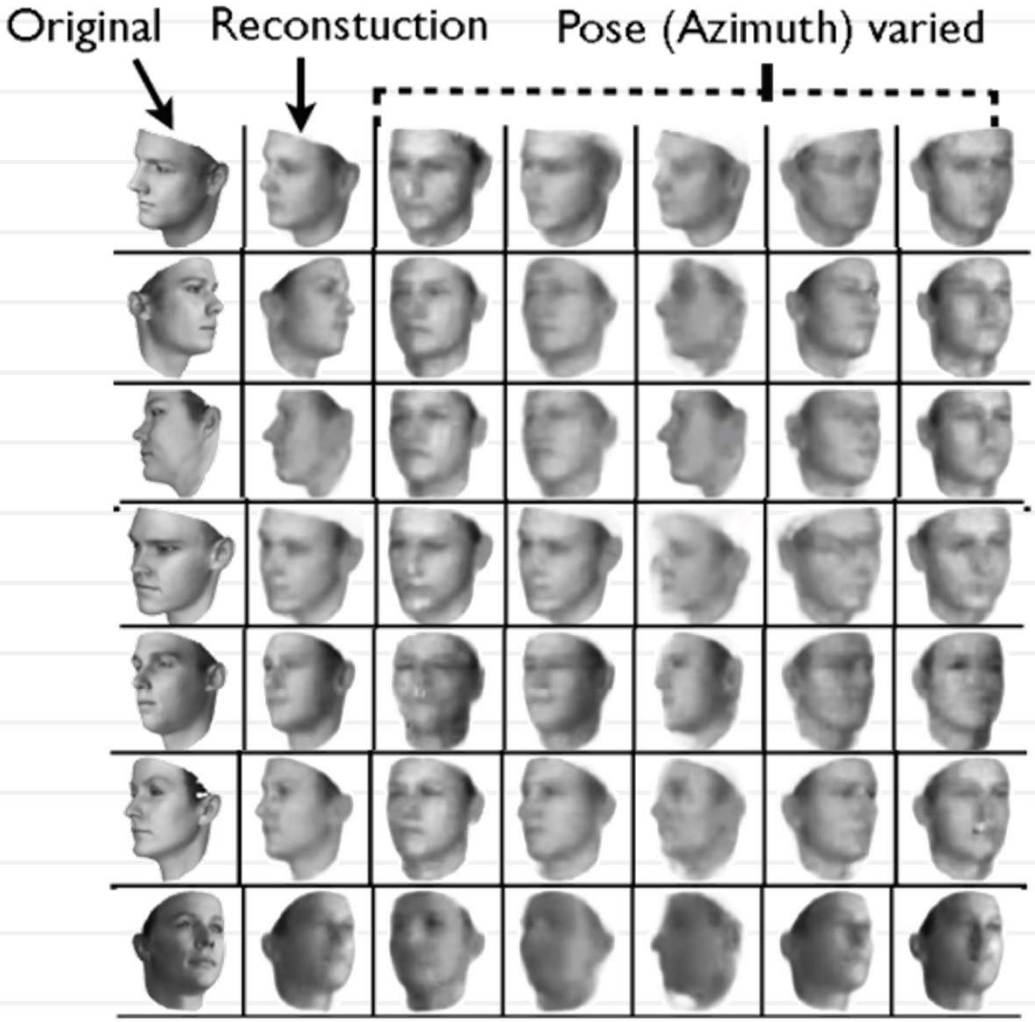


Training with clamping:

- Draw minibatches with some factors fixed
- During forwardprop replace corresponding z with averages $z_j^i = \bar{z}_j$
- During backprop replace loss gradients with $\frac{\partial L}{\partial z_j^i} = z_j^i - \bar{z}_j$

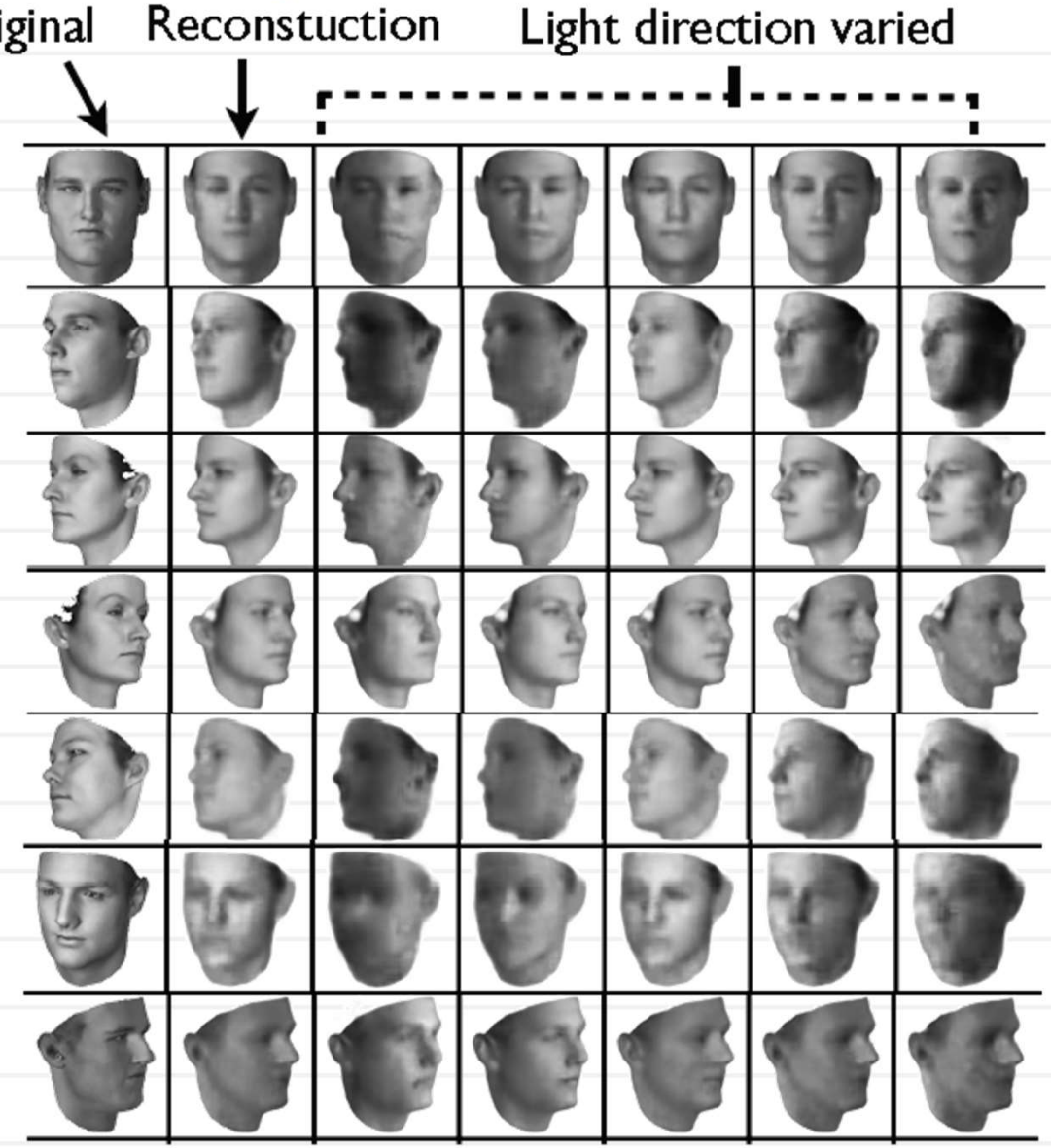
Deep Inverse Graphics





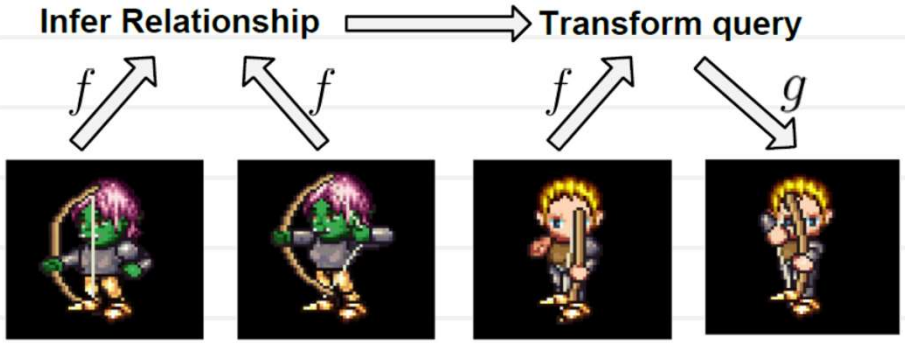
[Kulkarni et al. NIPS 2015]

Deep Inverse Graphics

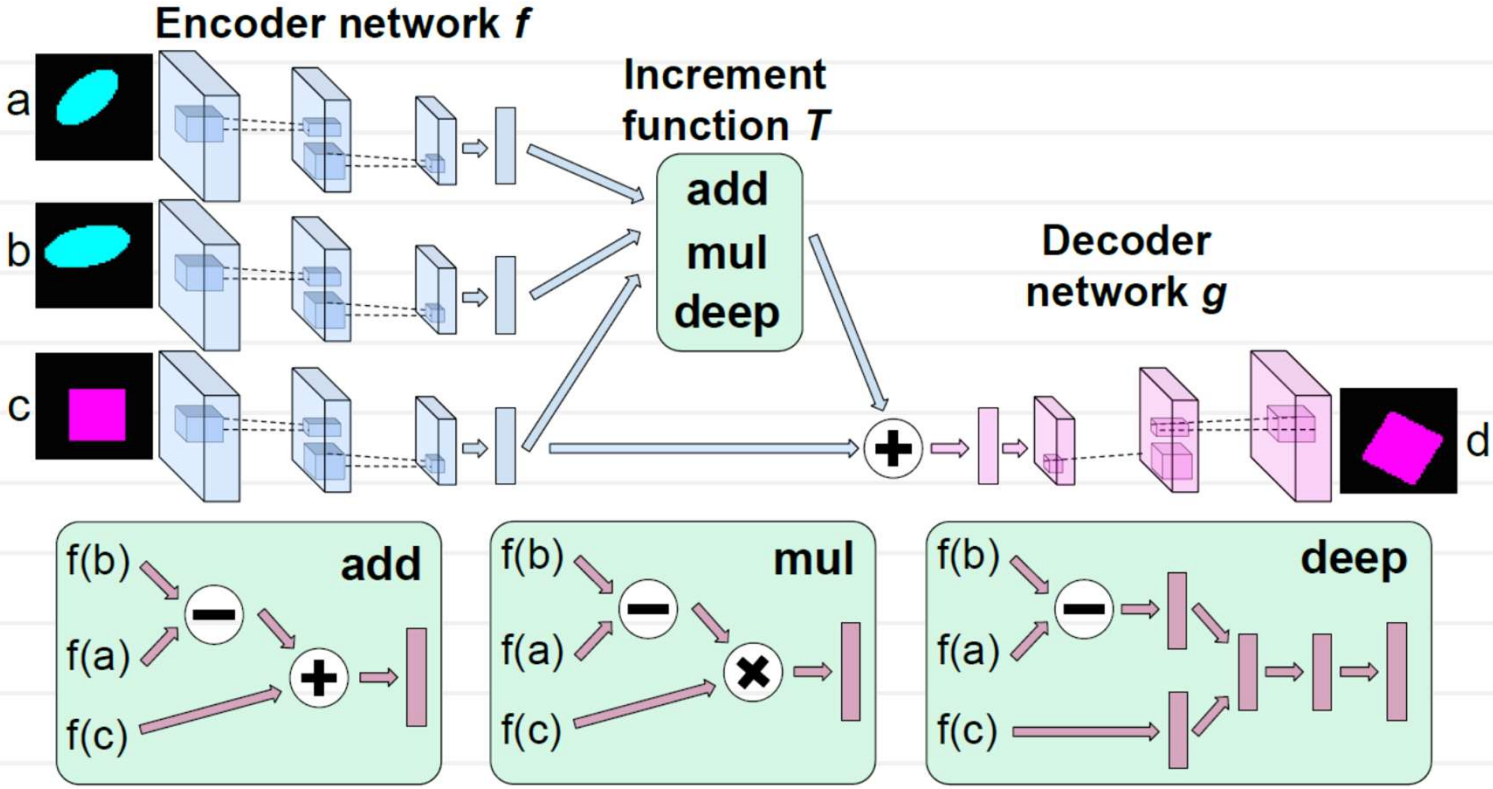


[Kulkarni et al. NIPS 2015]

"Deep visual analogy making"

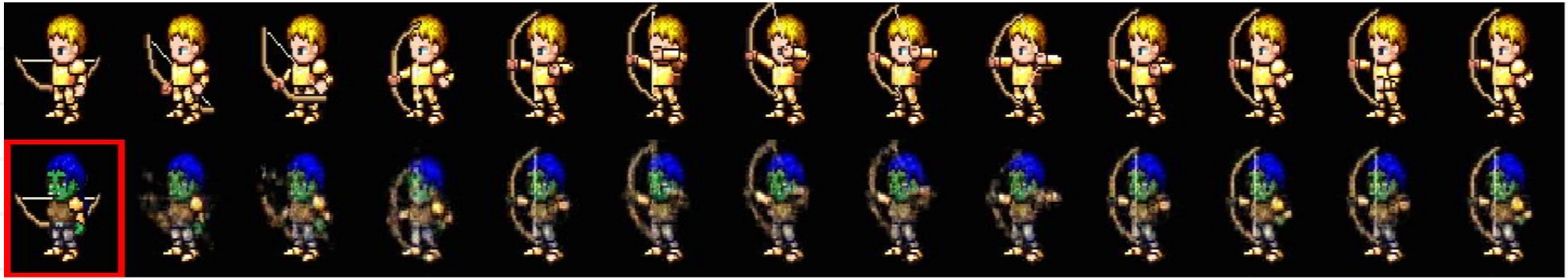


Fully-supervised setting:



[Reed et al. NIPS 2015]

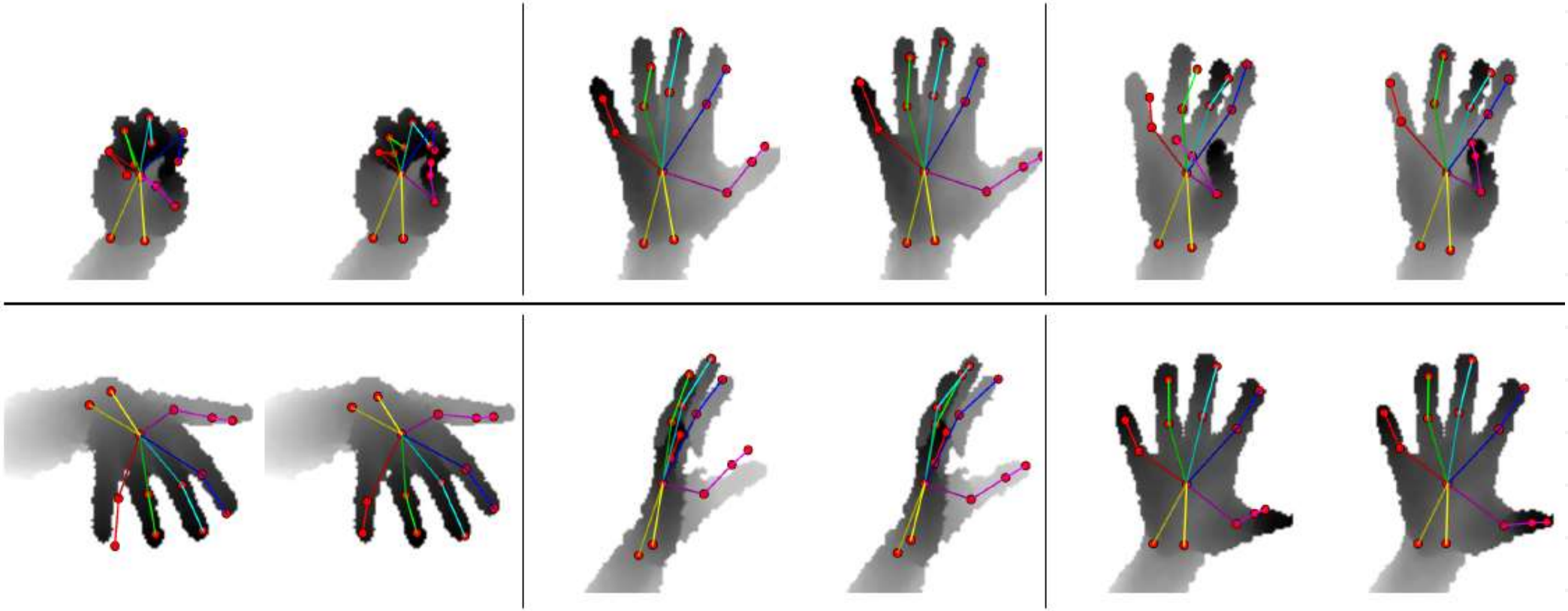
"Deep visual analogy making"



see also [Reed et al. NIPS 2015]

Vision with Feedback loop: background

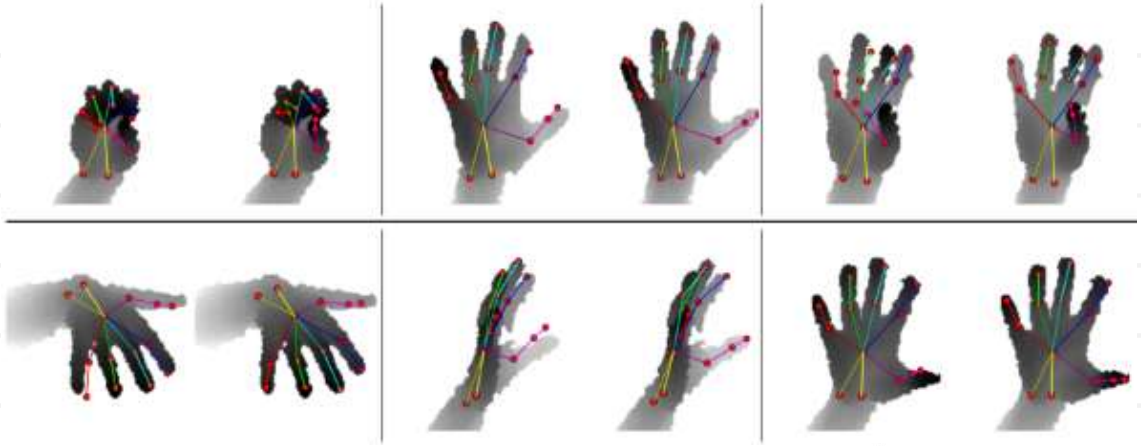
Problem setting:



- Difficult, even humans far from perfect
- Synthetic data is reasonable

[Oberweger, Wohlhart, Lepetit ICCV15]

Vision with Feedback loop : background

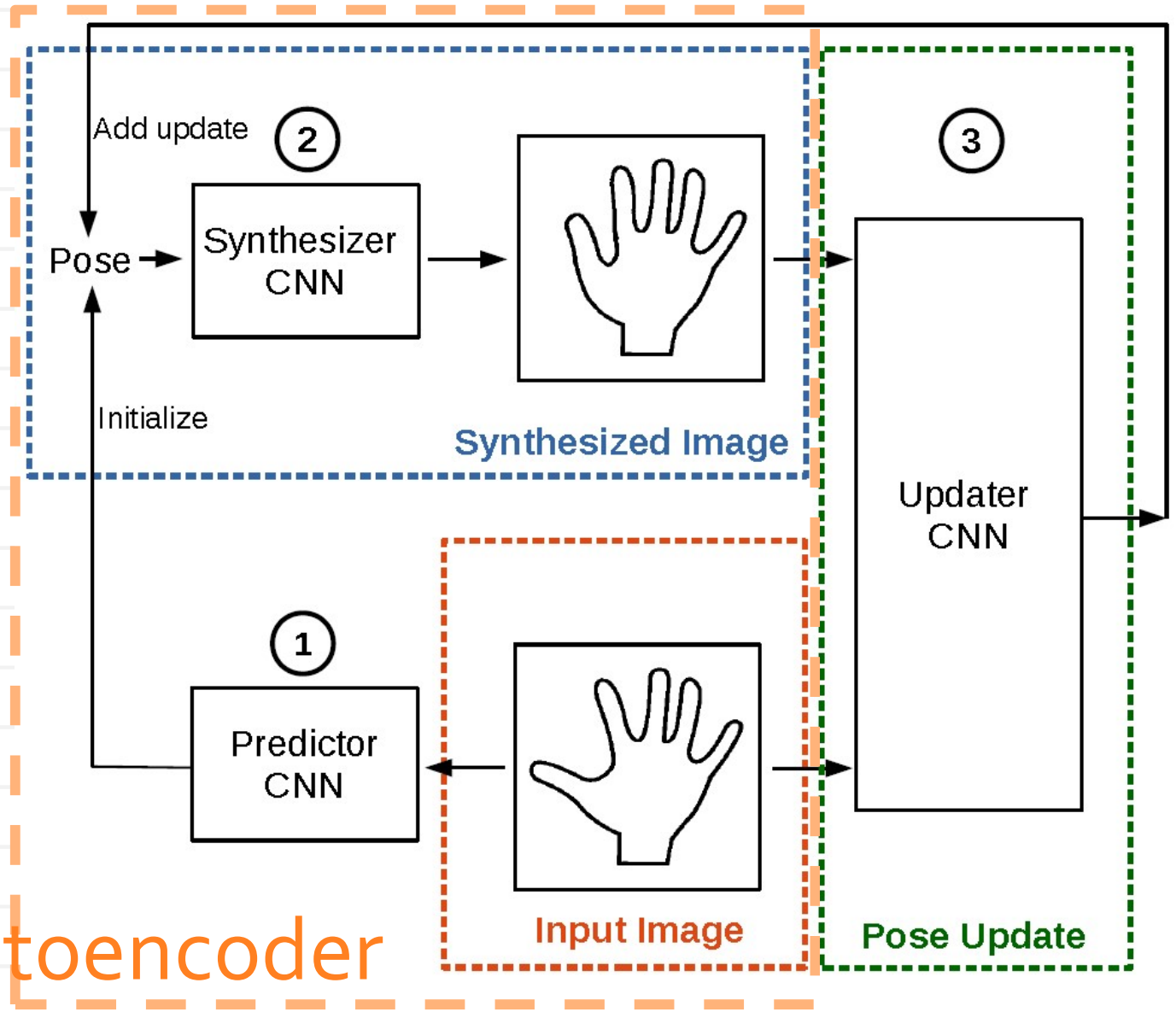


SoA: $\|I - R(p)\| \longrightarrow \min_p$

"gold standard algorithm"

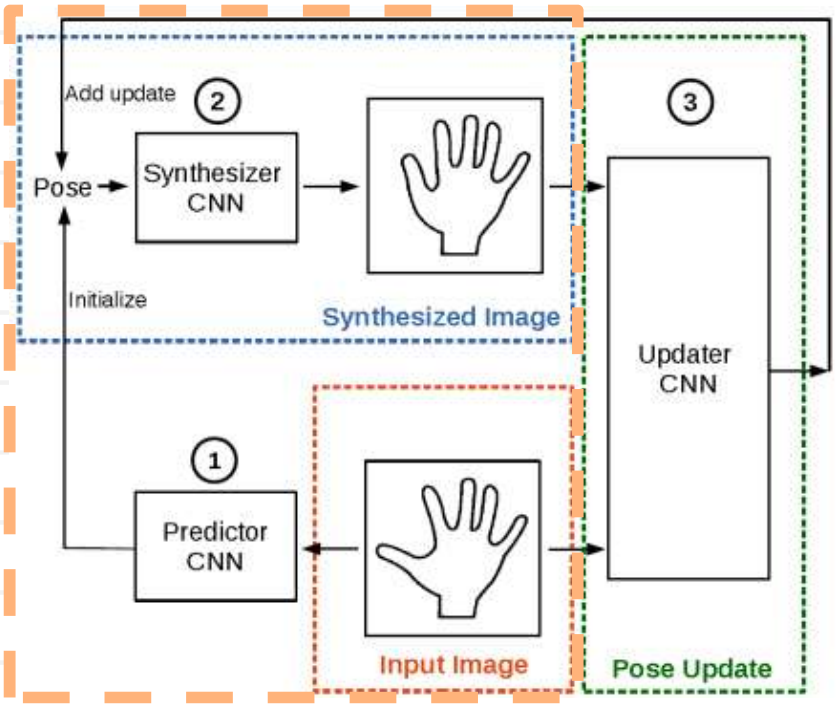
- Slow
- Get stuck in poor minima
- Needs good model
- When properly engineered, very hard to beat

Vision with Feedback loop



[Oberweger, Wohlhart, Lepetit ICCV15]

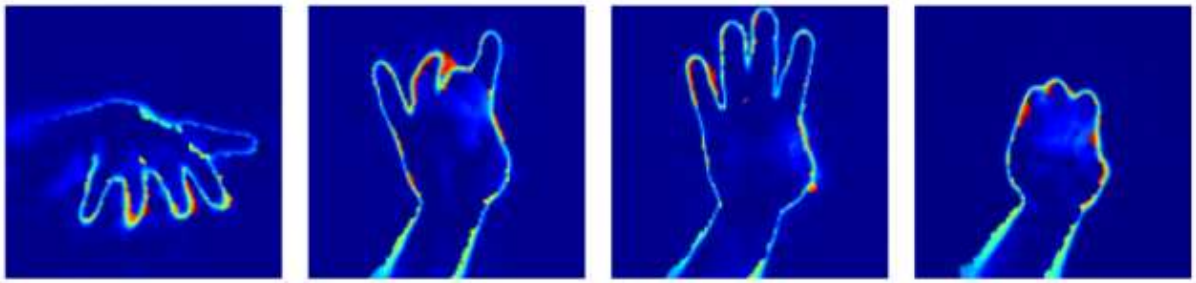
Vision with Feedback loop



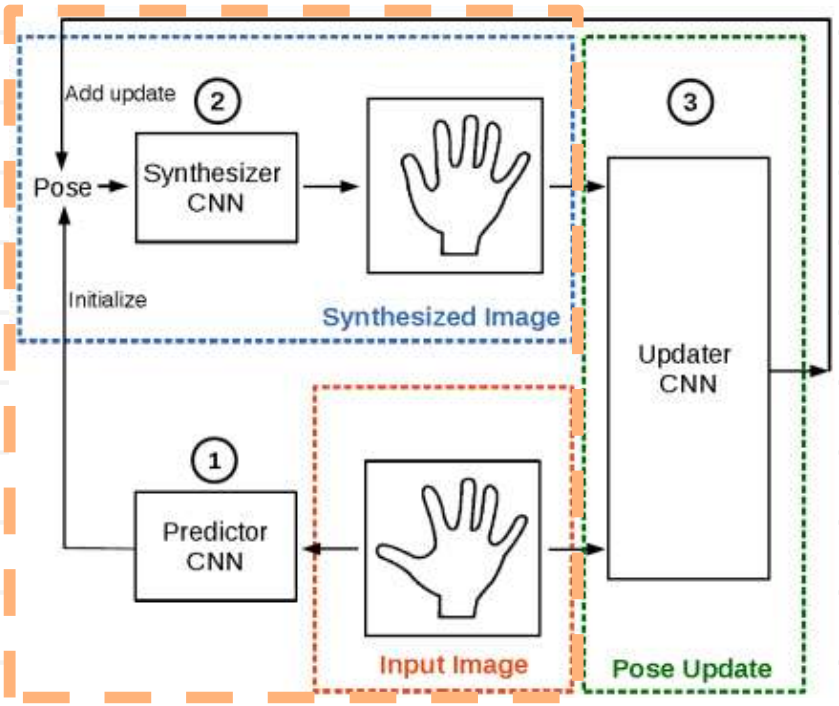
[Oberweger et al. ICCV15]

Autoencoder reconstruction:

Difference:

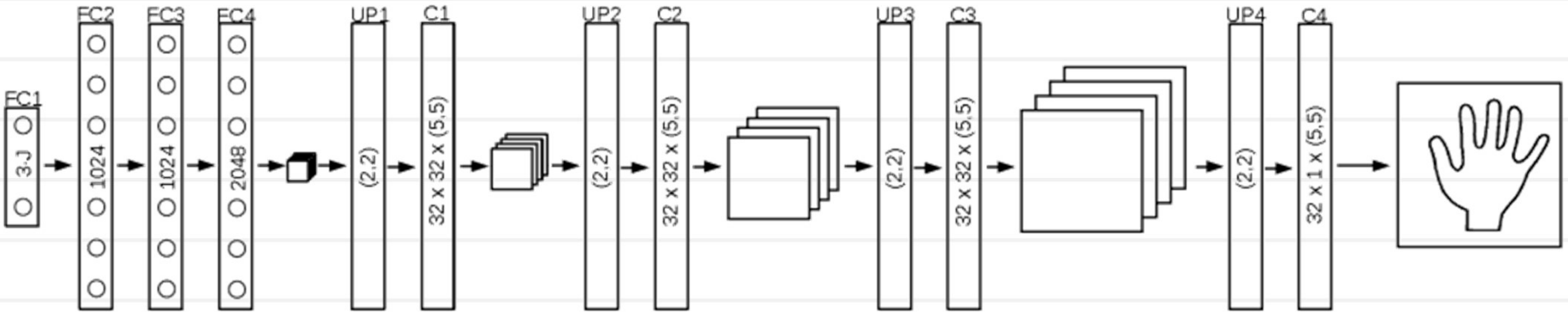


Vision with Feedback loop

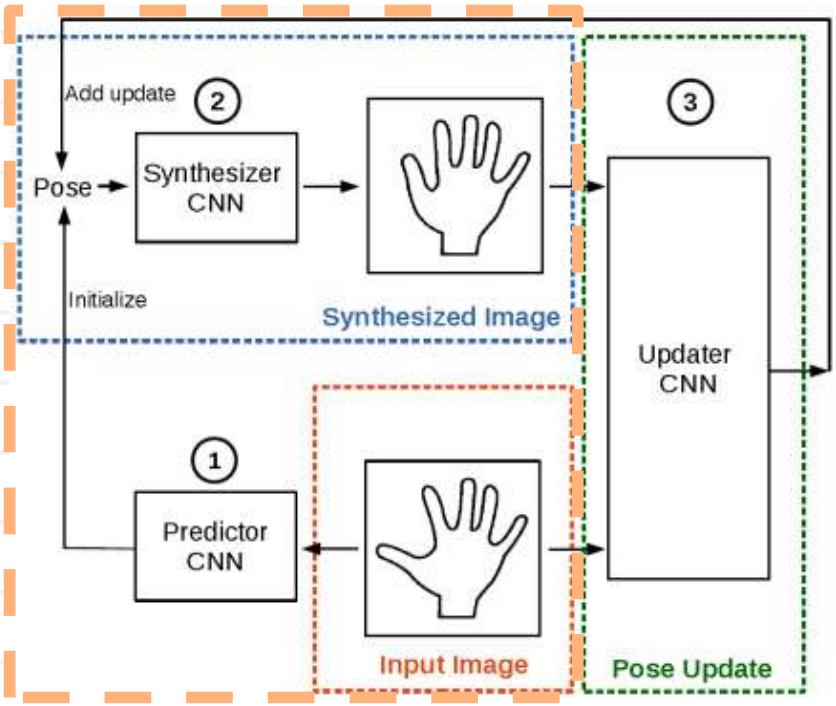


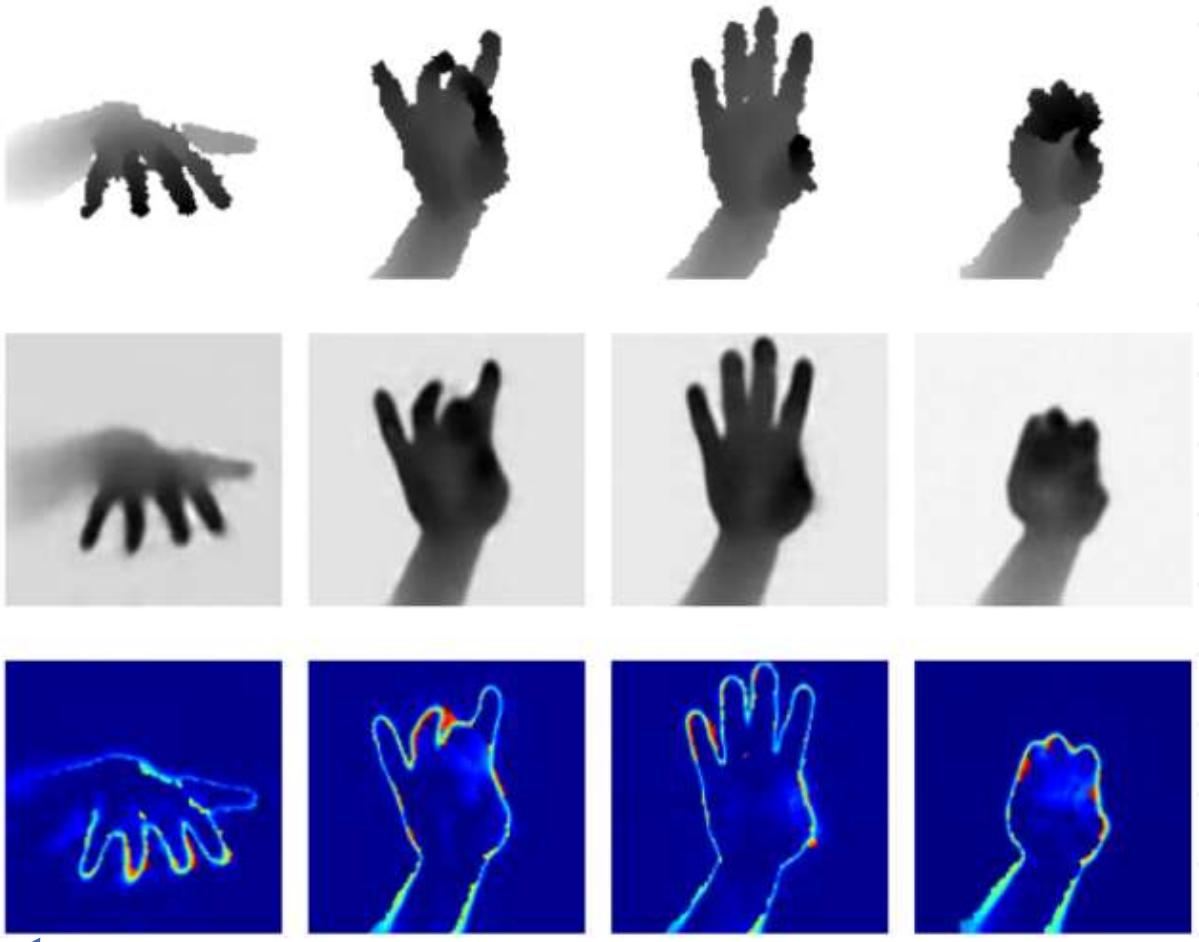
[Oberweger et al. ICCV15]

Synthesizer (decoder):



Vision with Feedback loop

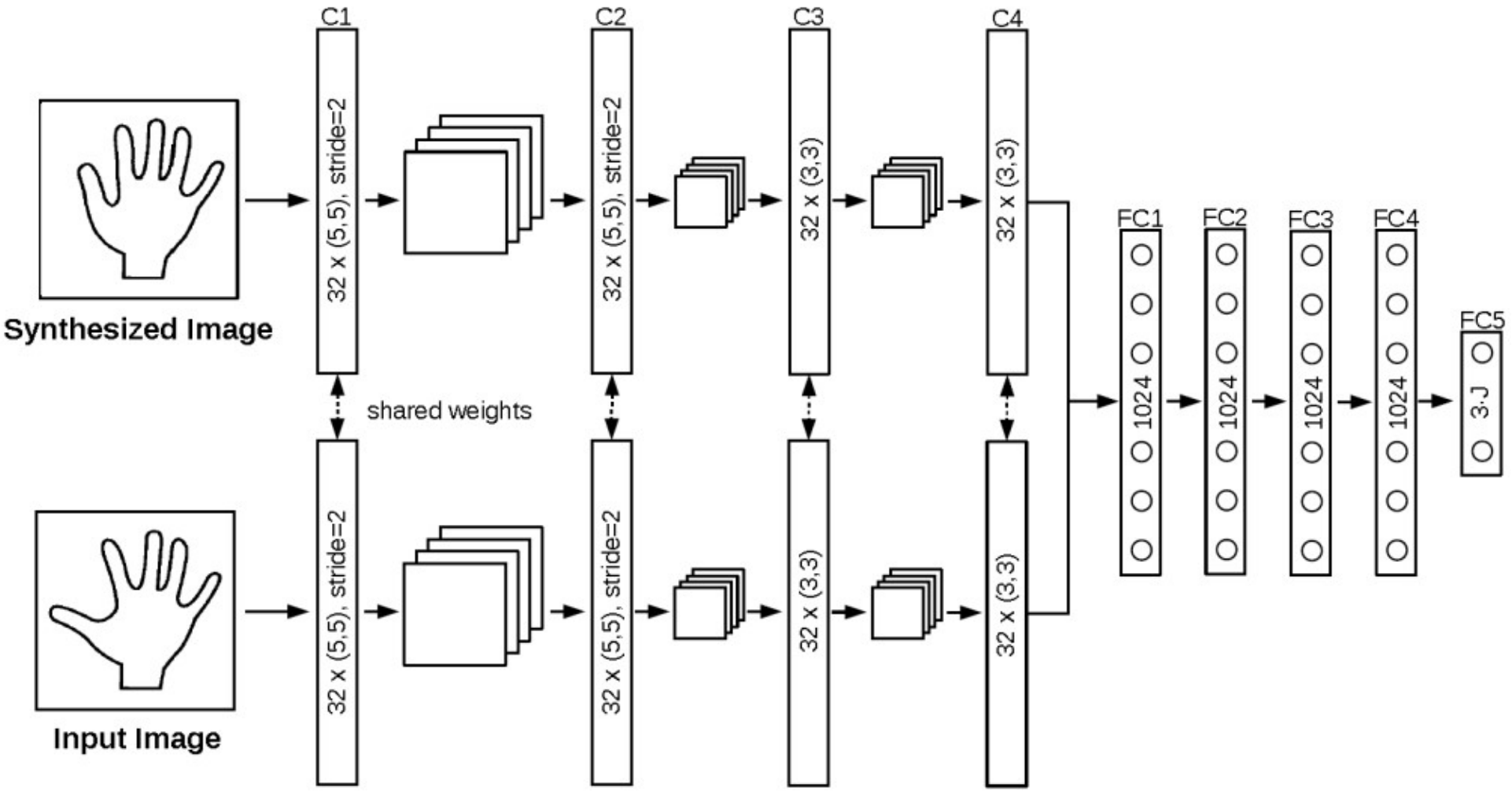




Where to improve, how to improve

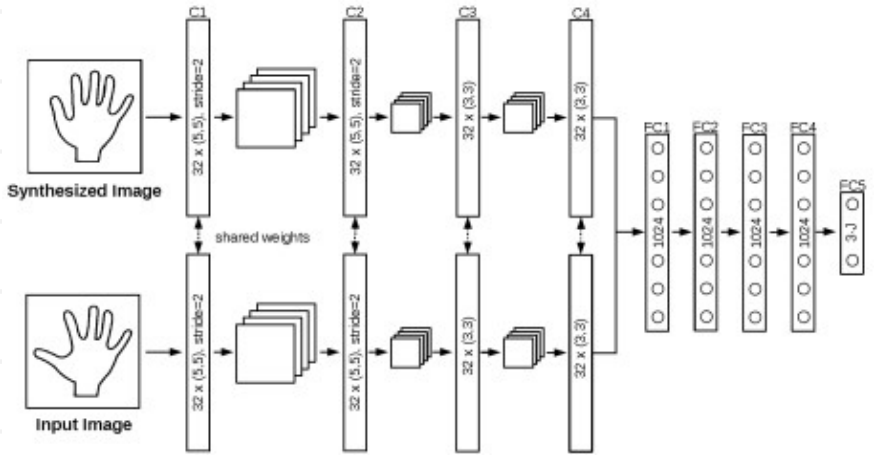
[Oberweger, Wohlhart, Lepetit ICCV15]

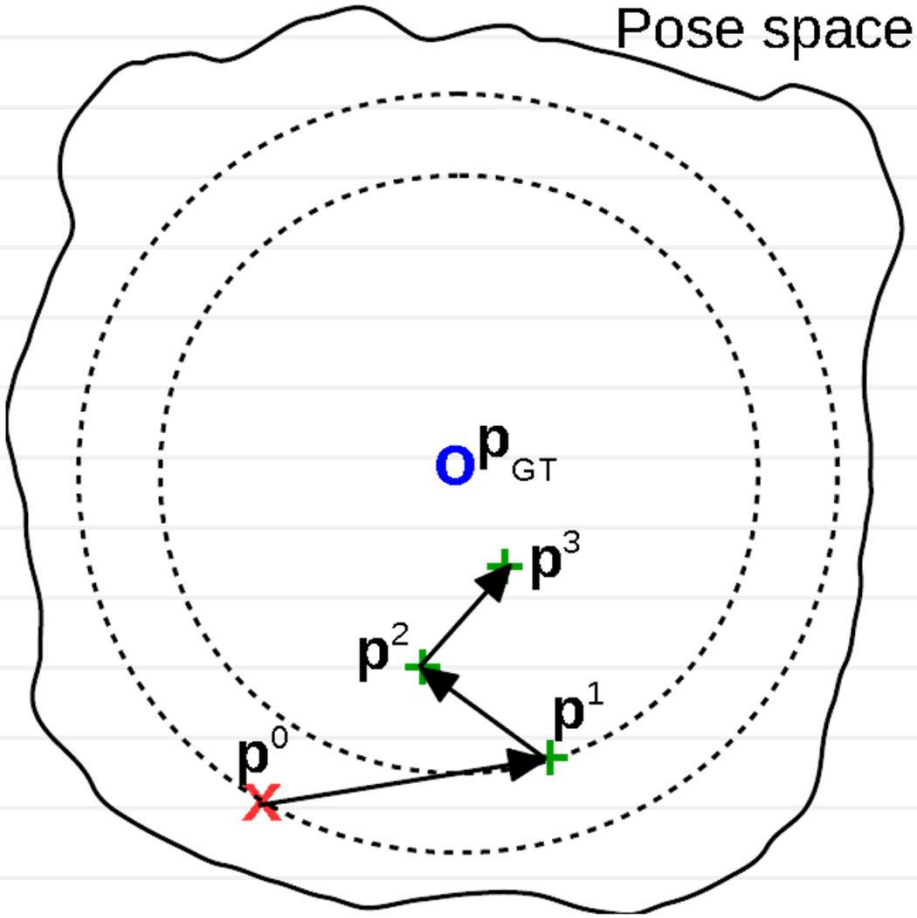
Updater network



[Oberweger, Wohlhart, Lepetit ICCV15]

Updater network



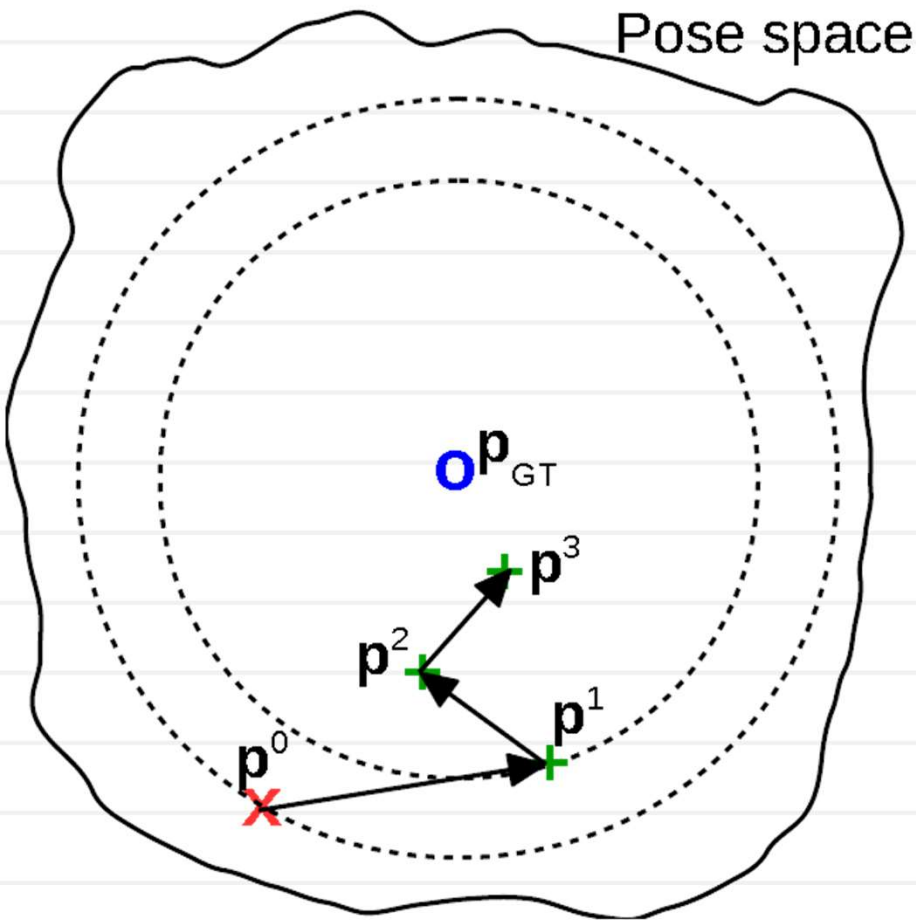


- Training with L2 loss does not work (too hard)
- Instead:

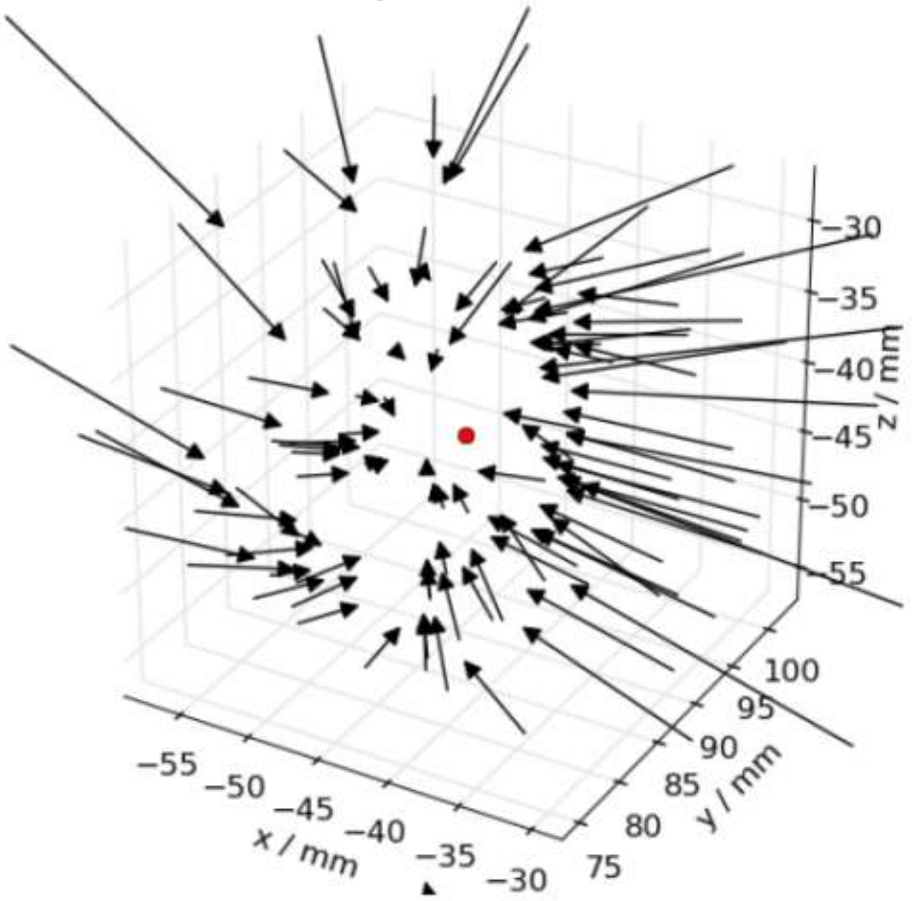
$$\mathcal{L} = \sum_{(\mathcal{D},\mathbf{p})\in\mathcal{T}} \sum_{\mathbf{p}'\in\mathcal{T}_{\mathcal{D}}} \max(0, \|\mathbf{p}''-\mathbf{p}\|_2 - \lambda\|\mathbf{p}'-\mathbf{p}\|_2)$$

[Oberweger et al.ICCV15]

Resulting updates



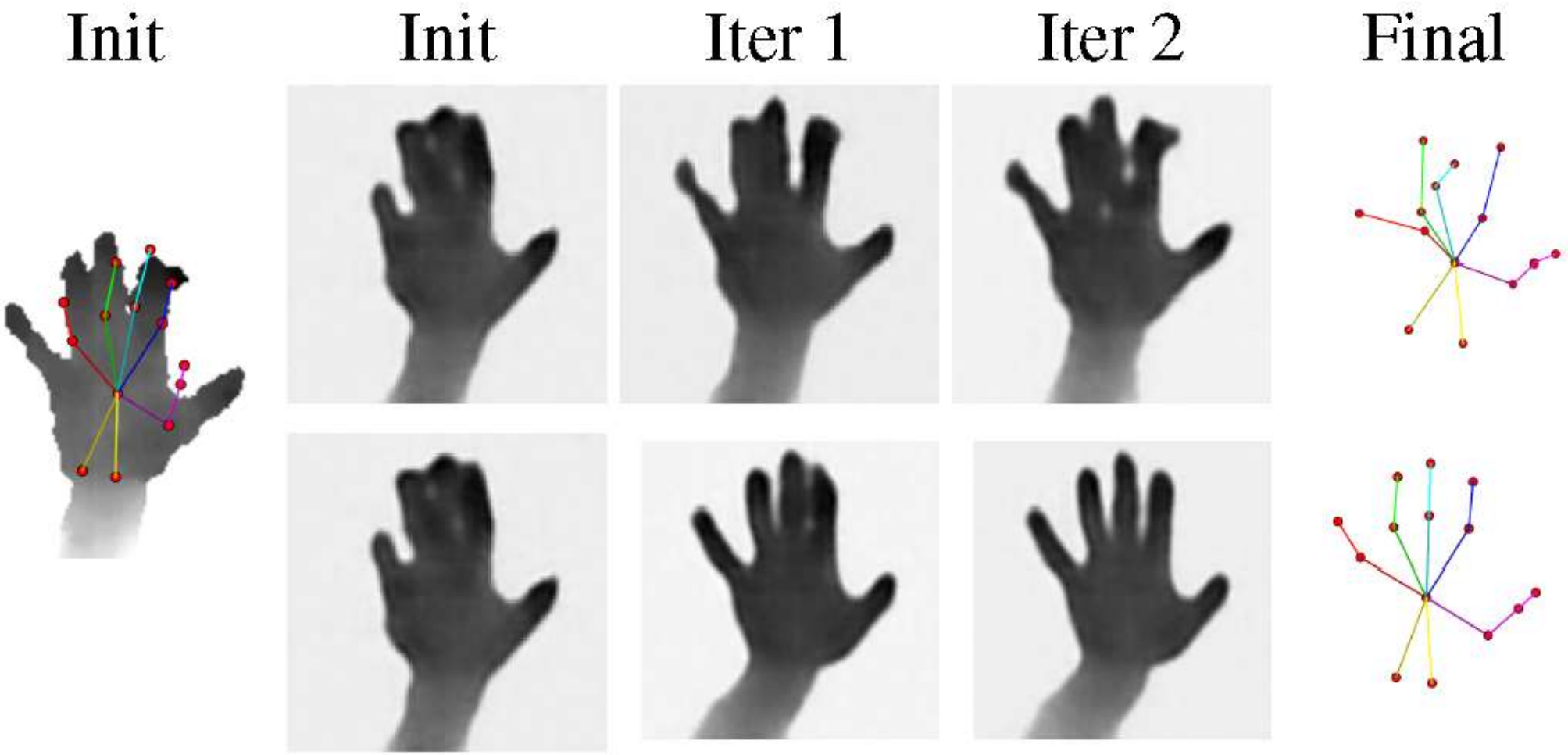
One of joints:



$$\mathcal{L} = \sum_{(\mathcal{D},\mathbf{p})\in\mathcal{T}} \sum_{\mathbf{p}'\in\mathcal{T}_{\mathcal{D}}} \max(0, \|\mathbf{p}''-\mathbf{p}\|_2 - \lambda\|\mathbf{p}'-\mathbf{p}\|_2)$$

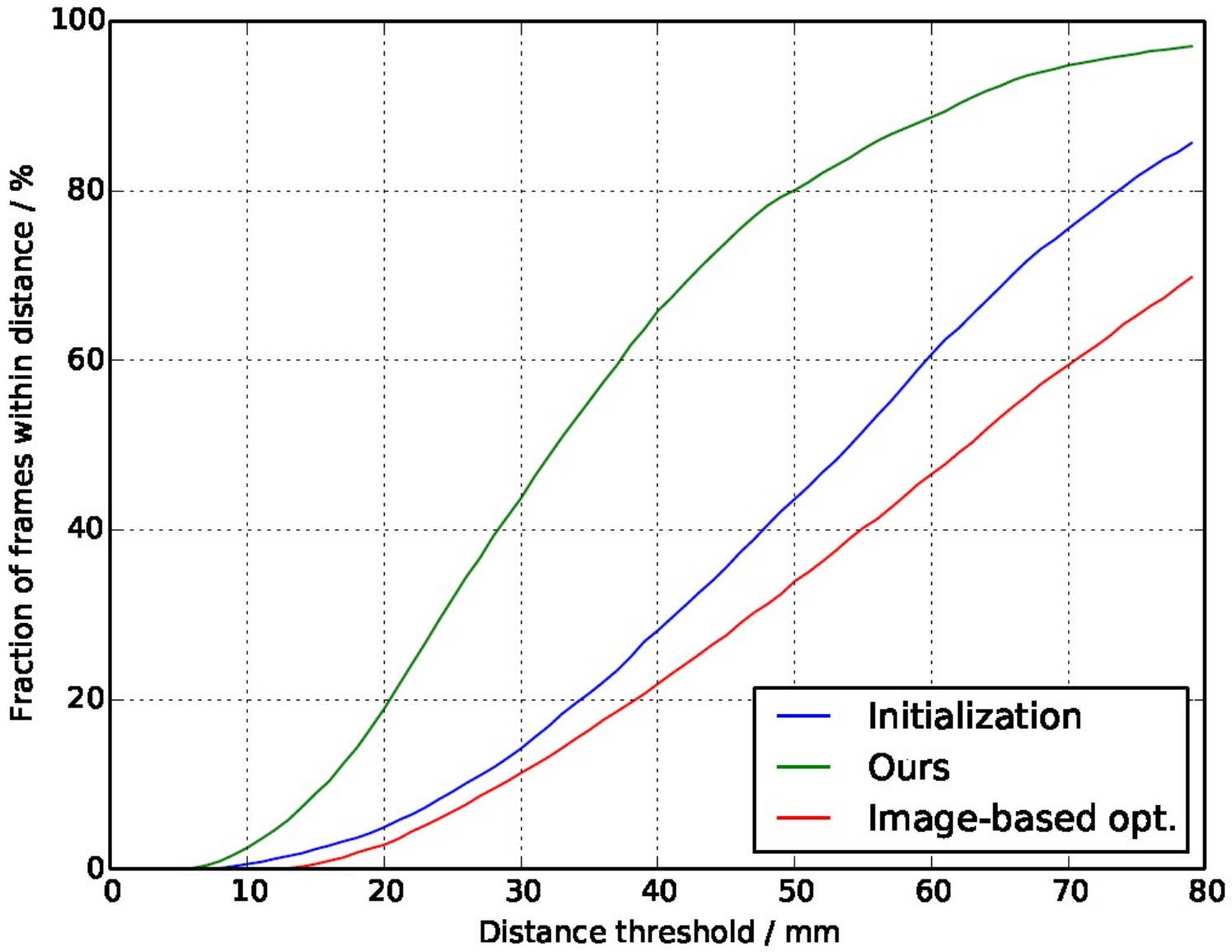
[Oberweger et al.ICCV15]

Qualitative comparison



[Oberweger, Wohlhart, Lepetit ICCV15]

Quantitative comparison



[Oberweger, Wohlhart, Lepetit ICCV15]

Recap

- Autoencoders are natural ways to tackle unsupervised learning with DL
- Natural ways to regularize: weight decay, contractive, denoising, *variational*
- Useful for pretraining if little annotated data
- Useful for generating and learning transforms
- Can be part of the bigger systems (c.f. last example)

Bibliography

Yoshua Bengio, Pascal Lamblin, Dan Popovici, Hugo Larochelle: Greedy Layer-Wise Training of Deep Networks. NIPS 2006: 153-160

Andrew Ng, Stanford class CS294a lecture notes, Handout #2

Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, Pierre-Antoine Manzagol: Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion. Journal of Machine Learning Research 11: 3371-3408 (2010)

Salah Rifai, Pascal Vincent, Xavier Muller, Xavier Glorot, Yoshua Bengio: Contractive Auto-Encoders: Explicit Invariance During Feature Extraction. ICML 2011: 833-840

Bibliography

Diederik P. Kingma, Max Welling:
Auto-Encoding Variational Bayes. ICLR 2014

Aaron Courville, Variational Autoencoders and Extensions, videolectures.net

Dumitru Erhan, Aaron C. Courville, Yoshua Bengio, Pascal Vincent:
Why Does Unsupervised Pre-training Help Deep Learning?
AISTATS 2010: 201-208

Tenenbaum, Joshua B., Vin De Silva, and John C. Langford. "A global geometric framework for nonlinear dimensionality reduction." Science 290.5500 (2000): 2319-2323.

Bibliography

Tejas D. Kulkarni, Will Whitney, Pushmeet Kohli, Joshua B. Tenenbaum: Deep Convolutional Inverse Graphics Network. NIPS 2015

Volker Blanz, Thomas Vetter:
A Morphable Model for the Synthesis of 3D Faces. SIGGRAPH 1999: 187-194

Reed, S.E., Zhang, Y., Zhang, Y. and Lee, H., et al. "Deep Visual Analogy-Making." NIPS 2015.

Markus Oberweger, Paul Wohlhart, Vincent Lepetit:
Training a Feedback Loop for Hand Pose Estimation. ICCV 2015: 3316-3324