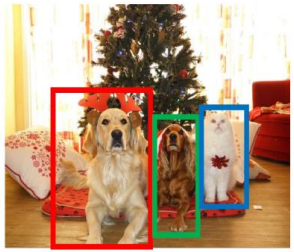
Deep Generative Model

Generative vs Discriminative models, Variational Autoencoders, Autoregressive models: RIDE, PixelRNN/CNN, PixelCNN++, Conditional Generative models

Success of Discriminative models

Cat

Classification



DOG, DOG, CAT

Object Detection

Semantic Segmentation

ImageNet

- 1 million labeled images

MS COCO, Pascal VOC

- 0.1 million images with object instances

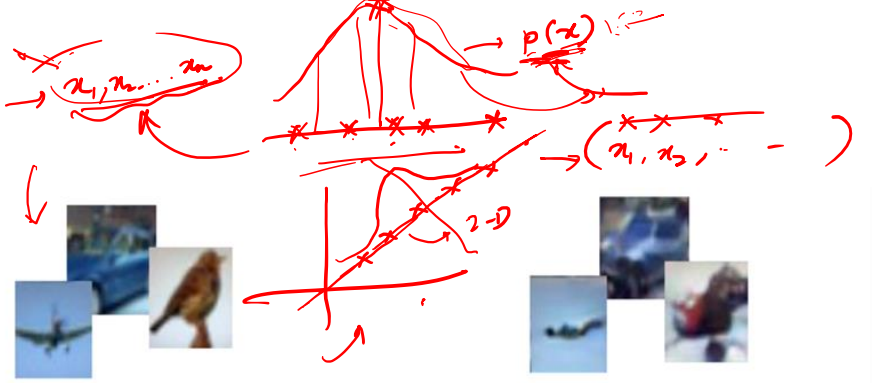
CityScapes

- 5k HR frames with pixel-wise labels

Need *tons* of data

Generative models

Goal: To model the data density, *p(x)*

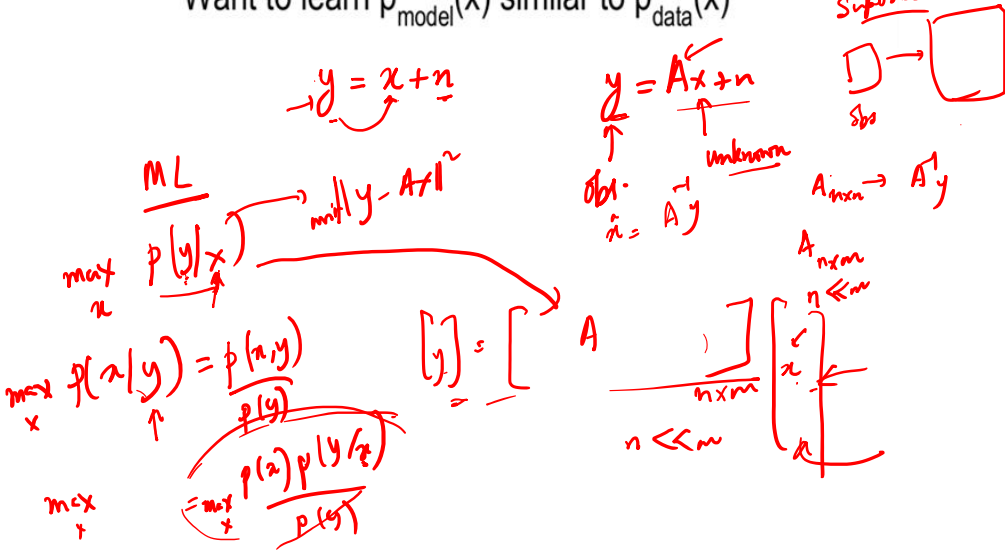


Training data ~ $p_{data}(x)$ Generated samples ~ $p_{model}(x)$

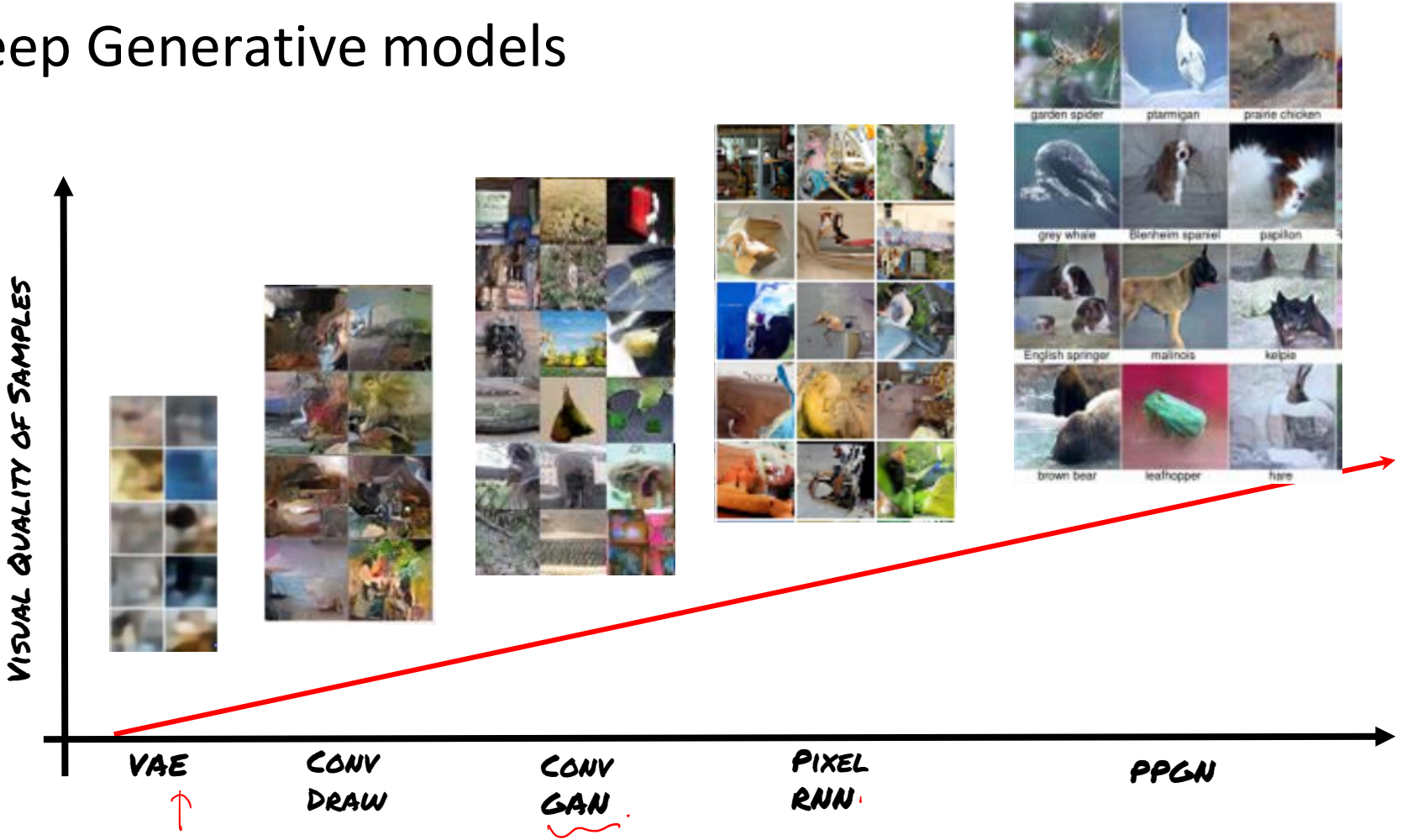
So, **Why** model *p(x)*?

- To improve generalisation
- To create synthetic training data
- Practical tasks like speech synthesis
- To simulate situations
- To understand the data **better**

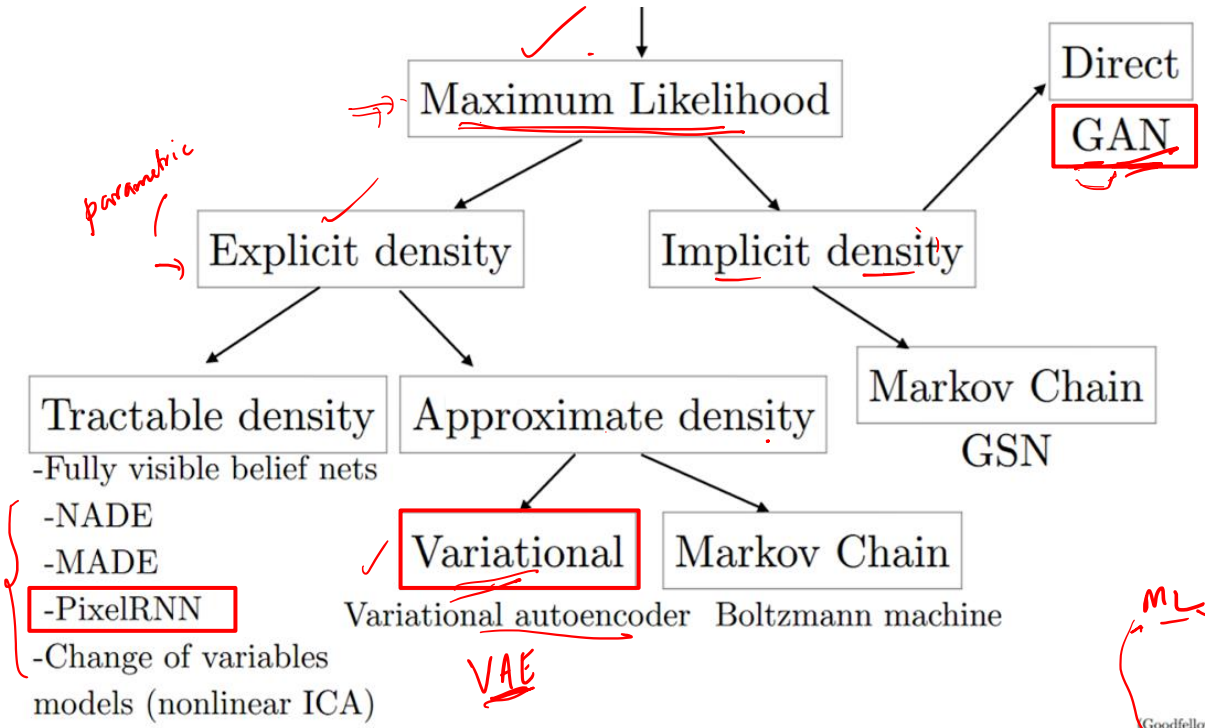
Want to learn $p_{model}(x)$ similar to $p_{data}(x)$

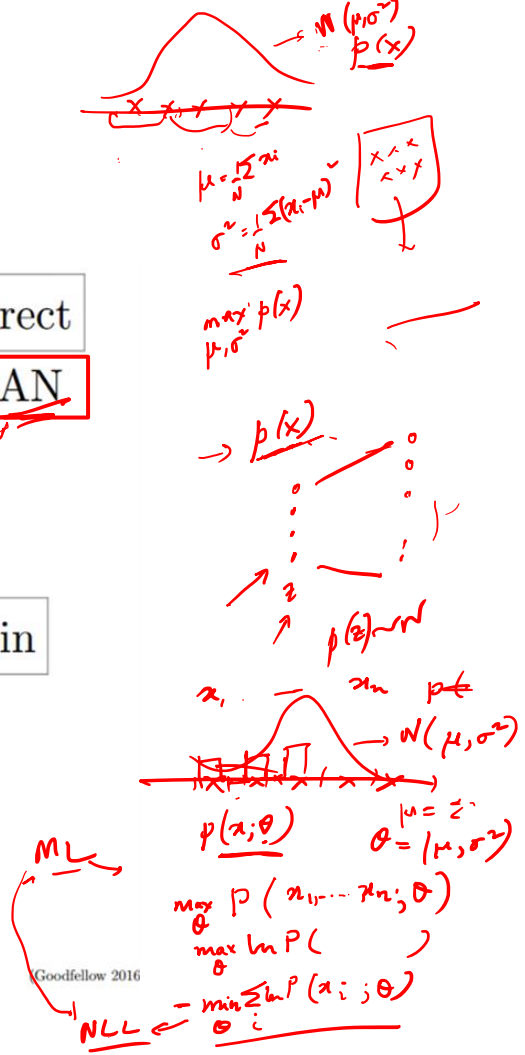


Deep Generative models



What are we going to look at

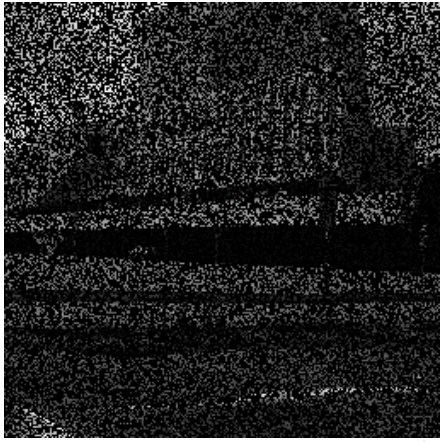




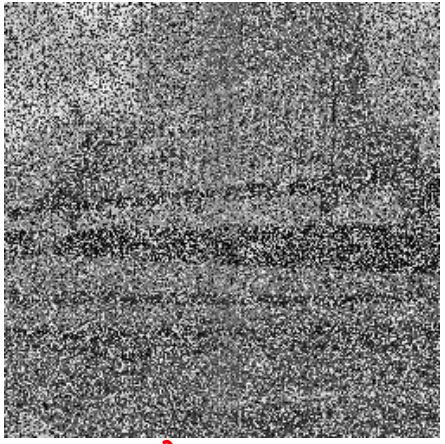
Applications: Solving inverse problems

Original Image

70% missing pixels



During inference



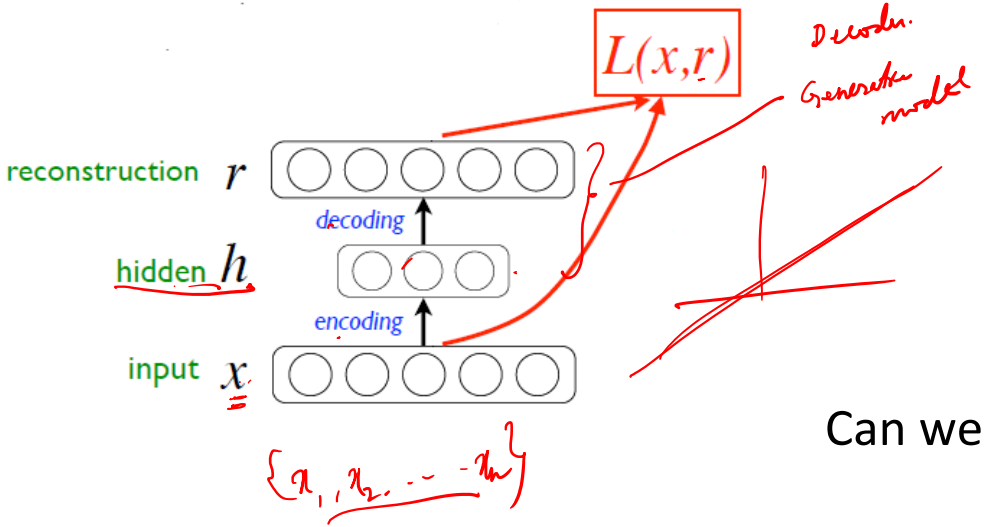
Final inpainted image

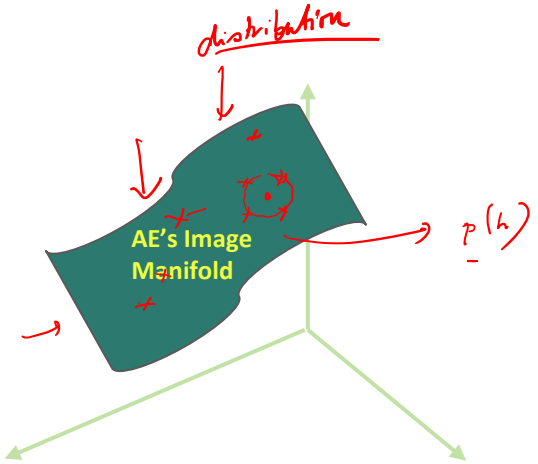
Applications

Variational Auto Encoder (VAE)

$p(x)$

Autoencoders

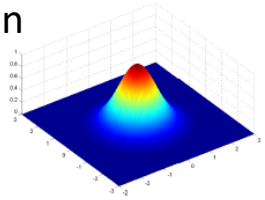


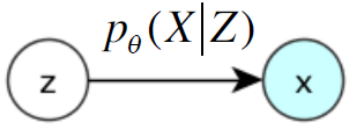


Can we generate samples from this manifold?

Training a Decoder to sample

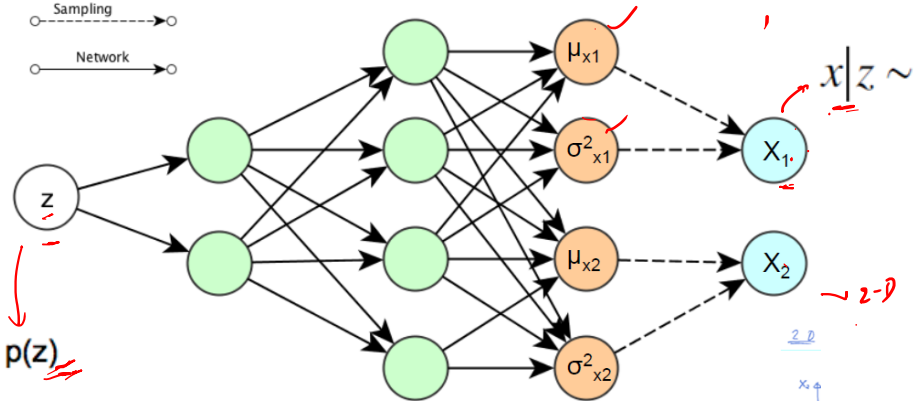
- Specify a distribution on latent variable P(z)
  - P(z) could be multivariate Gaussian



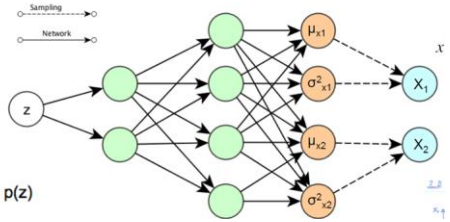


One Example

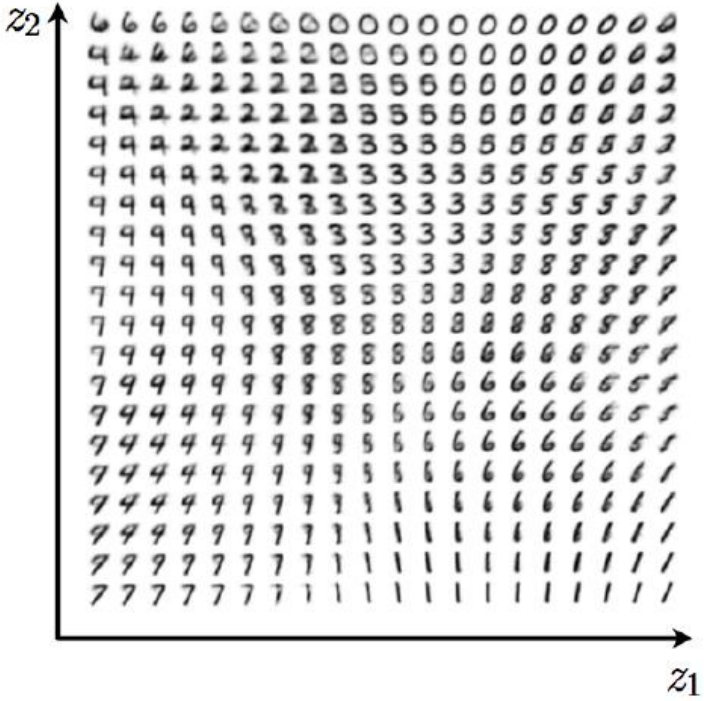
- Learn a decoder distribution P(x|z)



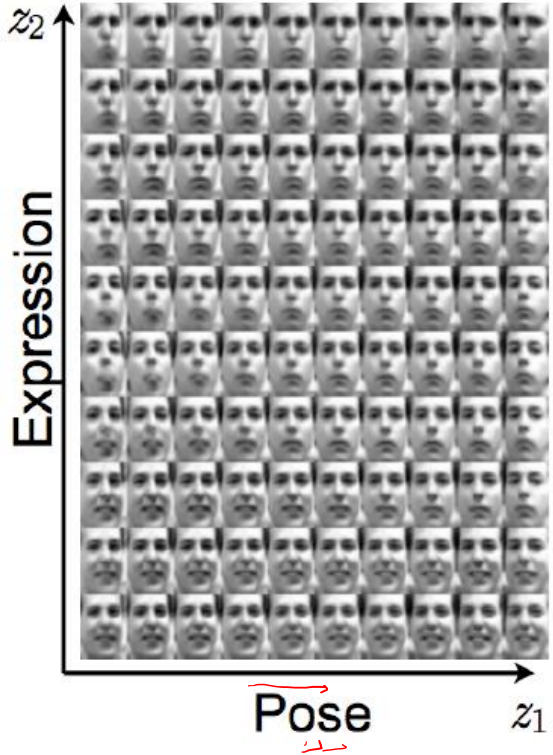
What z, latent code, can learn?



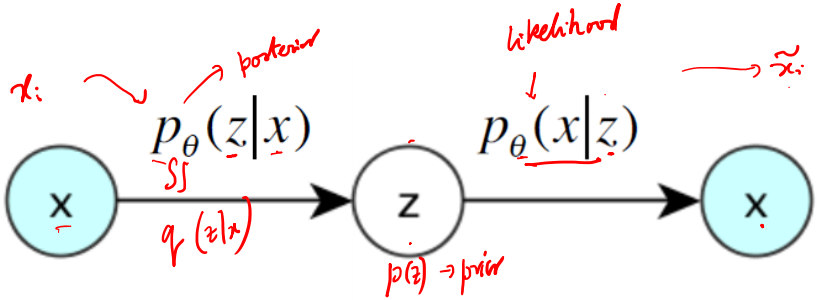
MNIST:

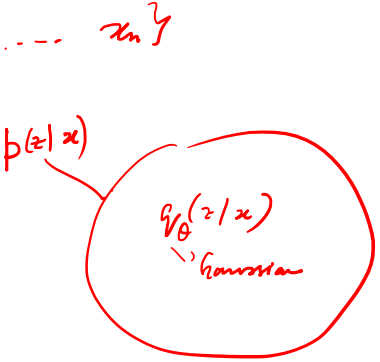


Frey Face dataset:



Where does z, latent code, come from?



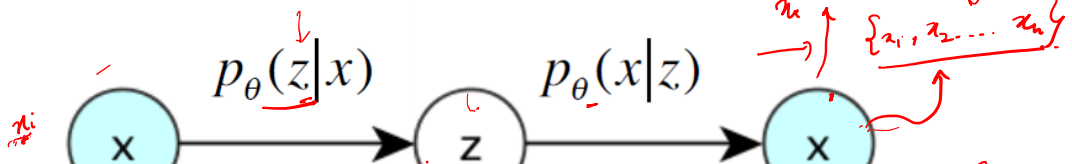


- Marginal, p(x), is intractable for complex distributions
- Solution: approximate p(z|x) by q(z|x) using variational inference

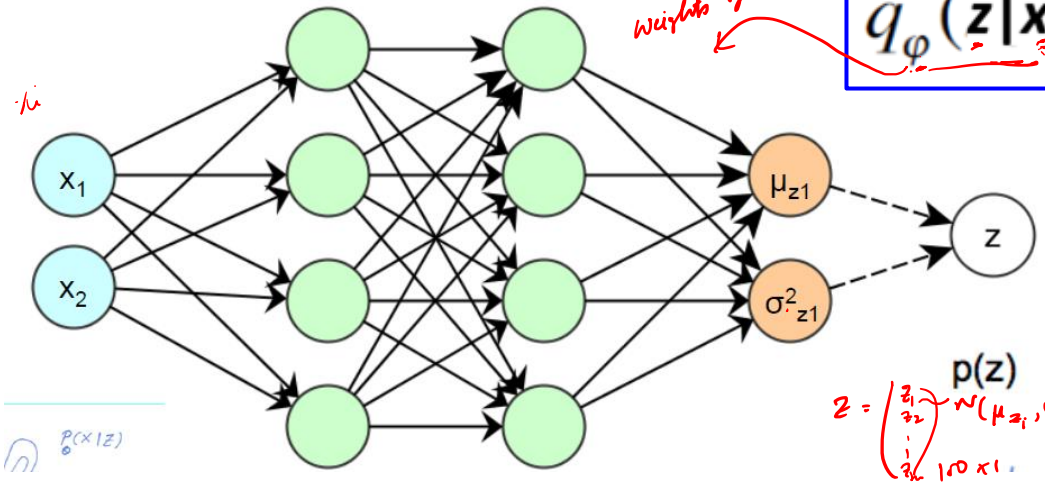
$$p(z|x) = \frac{p(x|z)p(z)}{\int p(x,z)dz}$$

likelihood prior posterior marginal

Approximate, posterior $p(z|x)$ using $q(z|x)$



$$q_\varphi(z|x) \sim p_\theta(z|x)$$



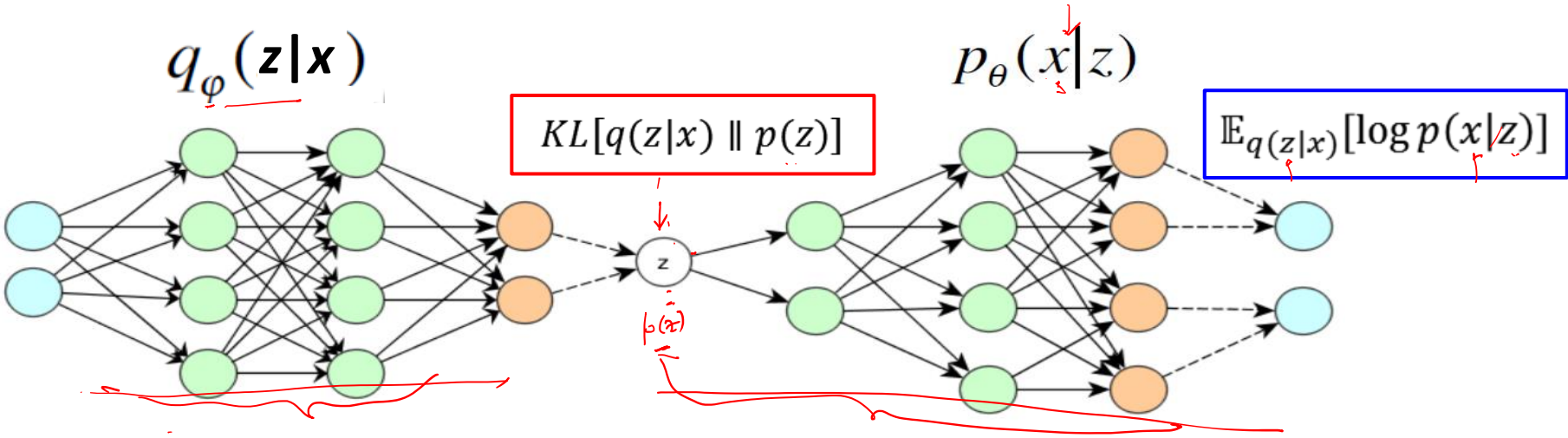
Variational Autoencoder (VAE)

- Find encoder and decoder parameters such that
  - If we sample from q(z|x), then the decoder should reconstruct back x
    - Maximize the expected log likelihood value $\mathbb{E}_{q(z|x)}[\log p(x|z)]$
  - q(z|x) should be close to p(z)
    - Minimize KL divergence $KL[q(z|x) \parallel p(z)]$

- Thus overall, maximize

$$\mathbb{E}_{q(z|x)}[\log p(x|z)] - KL[q(z|x) \parallel p(z)]$$

Variational Autoencoder (VAE)



$$\mathbb{E}_{q(z|x)}[\log p(x|z)] - KL[q(z|x) \parallel p(z)]$$

Autoregressive models

Autoregressive models: RIDE, PixelRNN/CNN, PixelCNN++, Conditional Generative models

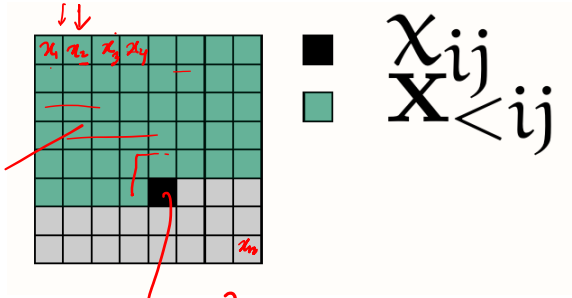
Factorization of a distribution

$p(x_1, x_2, x_3) = p(x_1)\, p(x_2|x_1)\, p(x_3|x_2,x_1)$

Is this the only valid factorization?

Let's look at one such factorization over an image, $\boldsymbol{x}$

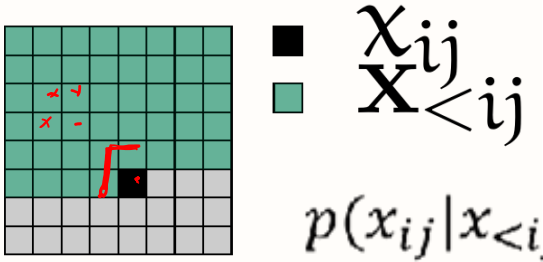
$$p(\mathbf{x}) = \prod_{i=1}^{n^2} p(x_i \mid x_1, \ldots, x_{i-1})$$



- If you look at the individual factor $p(x_i|x_{<i})$ - *conditional distribution*
- Modeling $p(x_i|x_{<i})$ is equivalent to sequence prediction,
- Current pixel x_i's distribution is modeled based on $x_{<i}$ causal context

Image courtesy: Theis et al., 2015

Modeling the image sequence ...



As you can see, depending on the position 'ij' the length of causal context, $x_{<ij}$, changes.

How to handle this?

- One way is to make Markovian assumption
- Model the distribution with Gaussian Scale Mixture (GSM)

$$p(x) = \int \varphi(z)\mathcal{N}(x;\mu,z\sigma)dz,$$

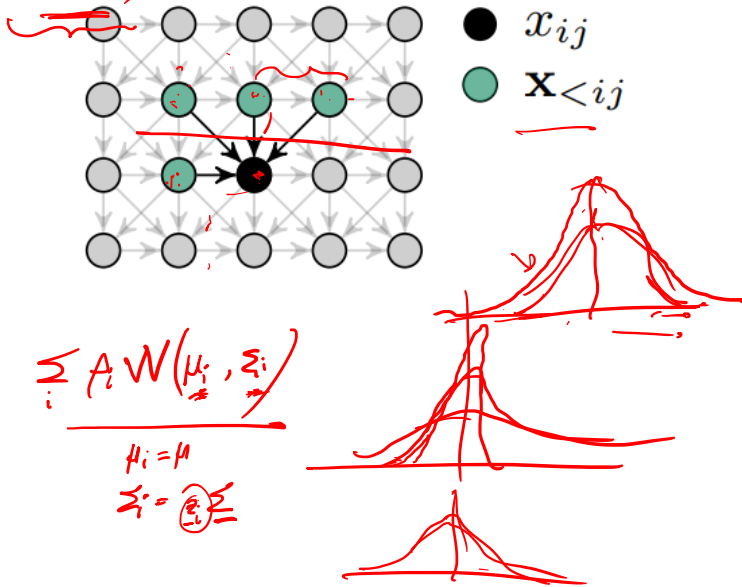
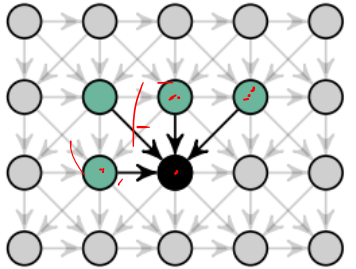


Image courtesy: Theis et al., 2015

Modeling the image sequence ...



- Markovian assumption
- Model the distribution with Mixture of Conditional GSM (MCGSM) by Theis et al. 2012

$$p(x_{ij}|\mathbf{x}_{<ij}, \boldsymbol{\theta}_{ij}) = \sum_{c,s} \underbrace{p(c, s|\mathbf{x}_{<ij}, \boldsymbol{\theta}_{ij})}_{\text{gate}} \underbrace{p(x_{i,j}|\mathbf{x}_{<ij}, c, s, \boldsymbol{\theta}_{ij})}_{\text{expert}}$$

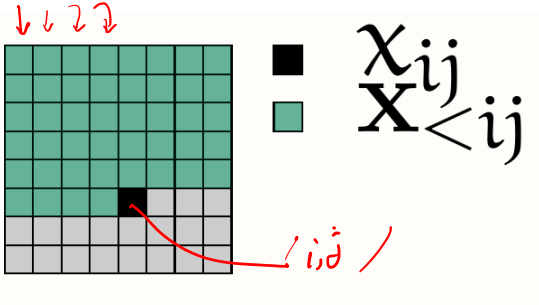
where the sum is over mixture component indices c corresponding to different covariances and scales s corresponding to different variances.

Note: length of causal context, $x_{<ij}$, is fixed

Slide courtesy: Theis et al., 2015

How to Train?

Factorization, $p(\mathbf{x}) = \prod_{ij} p(x_{ij}|x_{<ij})$



$$NLL: -\log p(\mathbf{x}) = -\sum_{ij} \log p(x_{ij}|x_{<ij}, \theta_{ij})$$

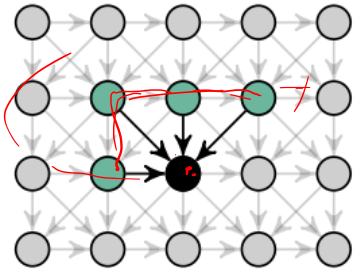
$$= -\sum_{ij} \log p(x_{ij}|x_{<ij}, \theta)$$

Parametrize, $p(x_{ij}|x_{<ij})$, conditional distribution with GMMs, GSMs ...

Training: Maximum likelihood

$$\underset{\theta}{argmin} -\sum_{n} \log p(\mathbf{x_n}, \boldsymbol{\theta})$$

Limitation

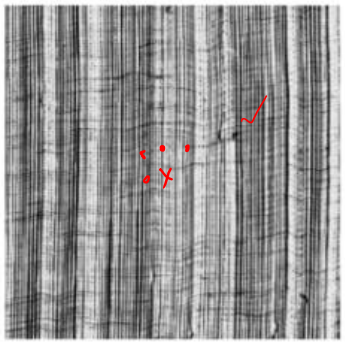


$$p(x_{ij}|\mathbf{x}_{<ij})$$

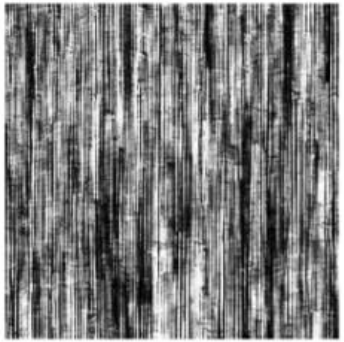
- MCGSM by Theis et al. 2012

$$p(x_{ij}|\mathbf{x}_{<ij}, \boldsymbol{\theta}_{ij}) = \sum_{c,s} \underbrace{p(c, s|\mathbf{x}_{<ij}, \boldsymbol{\theta}_{ij})}_{\text{gate}} \underbrace{p(x_{i,j}|\mathbf{x}_{<ij}, c, s, \boldsymbol{\theta}_{ij})}_{\text{expert}}$$

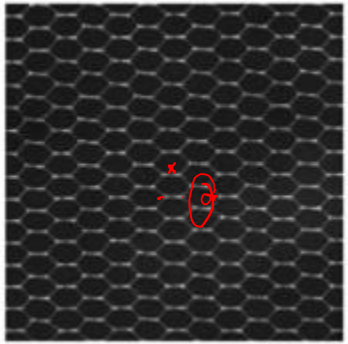
Fails to capture the long range dependencies in an image



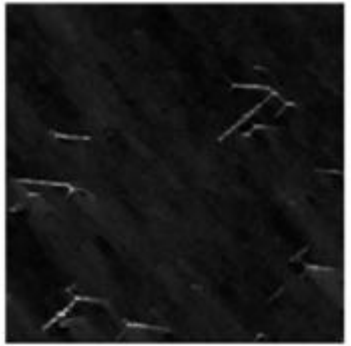
Actual image



Sample



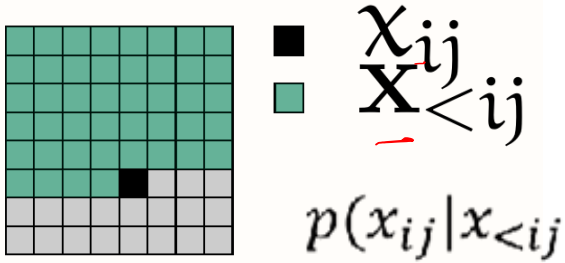
Actual image



Sample

Image courtesy: Theis et al., 2015

How to overcome this limitation?



As you can see, depending on the position 'ij' the length of causal context, $x_{<ij}$, changes.

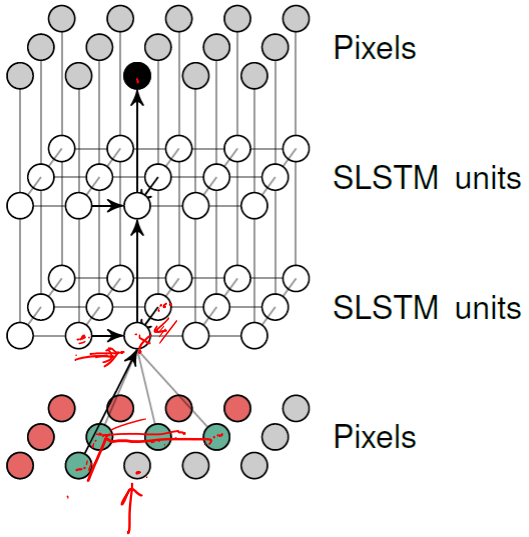
How to handle this?

- Markovian assumption - fails to capture long range dependencies
- Can we use RNNs to model them via $x_{<ij}$?

How do we use RNNs to model image sequence?

RIDE

- Recurrent Image Density Estimator by Theis et al. 2015
- SLSTM: a 2D Spatial LSTM summarizes the causal context, $x_{<ij}$ as h_{ij}



$$h_{ij} = f(x_{<ij}, h_{i-1,j}, h_{i,j-1})$$

$$\begin{pmatrix} \mathbf{g}_{ij} \\ \mathbf{o}_{ij} \\ \mathbf{i}_{ij} \\ \mathbf{f}^r_{ij} \\ \mathbf{f}^c_{ij} \end{pmatrix} = \begin{pmatrix} \tanh \\ \sigma \\ \sigma \\ \sigma \\ \sigma \end{pmatrix} T_{\mathbf{A},\mathbf{b}} \begin{pmatrix} \mathbf{x}_{<ij} \\ \mathbf{h}_{i,j-1} \\ \mathbf{h}_{i-1,j} \end{pmatrix},$$

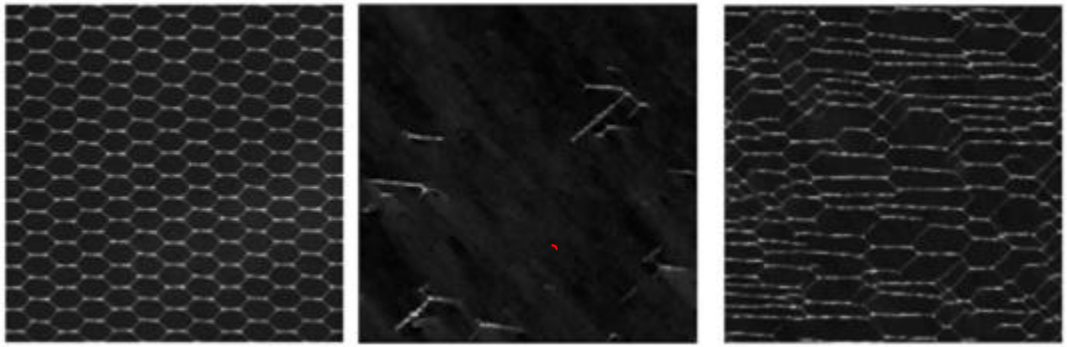
$$\mathbf{c}_{ij} = \mathbf{g}_{ij} \odot \mathbf{i}_{ij} + \mathbf{c}_{i,j-1} \odot \mathbf{f}^c_{ij} + \mathbf{c}_{i-1,j} \odot \mathbf{f}^r_{ij},$$

$$\mathbf{h}_{ij} = \tanh\left(\mathbf{c}_{ij} \odot \mathbf{o}_{ij}\right),$$

$$p(x_{ij}|h_{<ij}, \theta) = \sum_{c,s} p(c, s|h_{<ij}, \theta)\, p(x_{ij}|c, s, h_{<ij}, \theta)$$

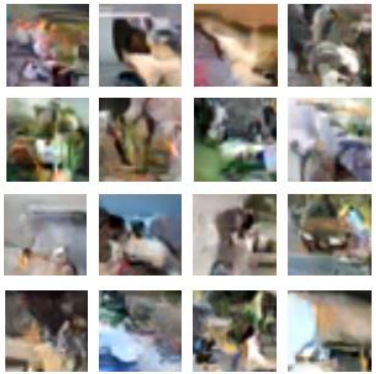
Image courtesy: Theis et al., 2015

RIDE



Actual image MCGSM sample RIDE sample

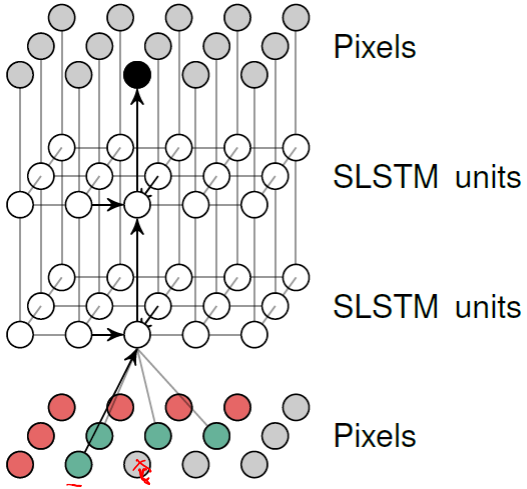
SLSTM improves in modeling *long-term* dependencies



RIDE samples trained on CIFAR dataset

RIDE summary

- SLSTM captures long-range dependencies in images well
- Sampling and training is time consuming because of sequential nature



Using Generative Models for Image Processing

Image inpainting, Compressive imaging, LiSens and FlatCam

Discriminative models - learns a mapping

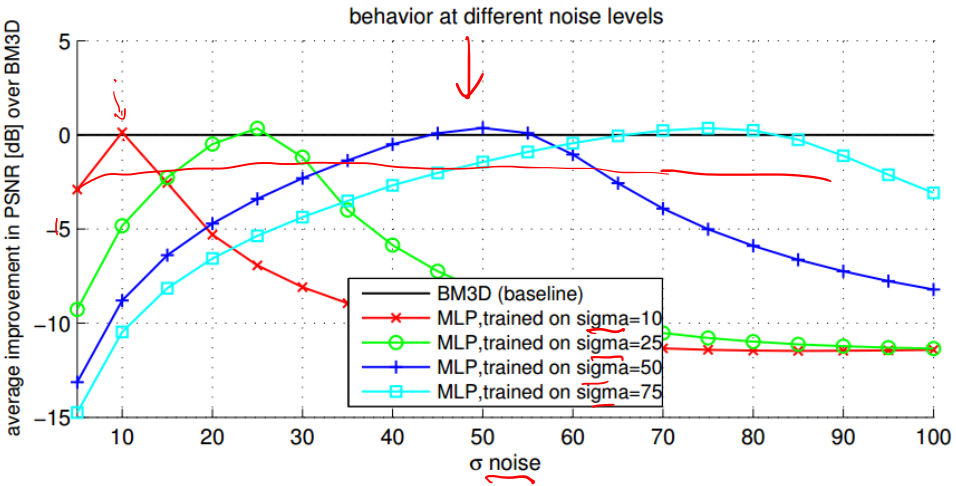
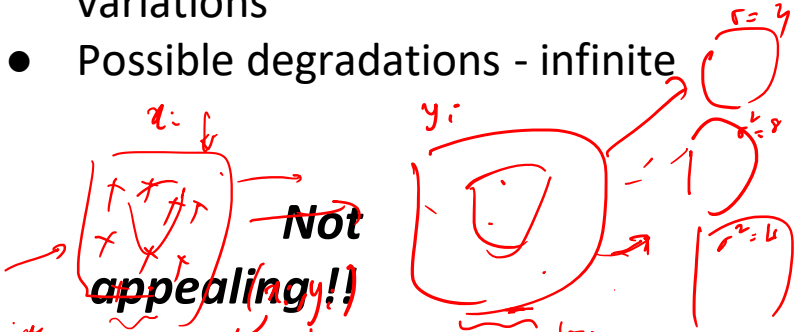


Image denoising using MLPs by HC Burger et al., 2012

- Learn a network - for each degradation
- Model may not generalize to variations
- Possible degradations - infinite



Not appealing!!

Generative modeling - *versatility*

$$Y = AX + n$$

Corrupted Image

Clean Image

MAP inference

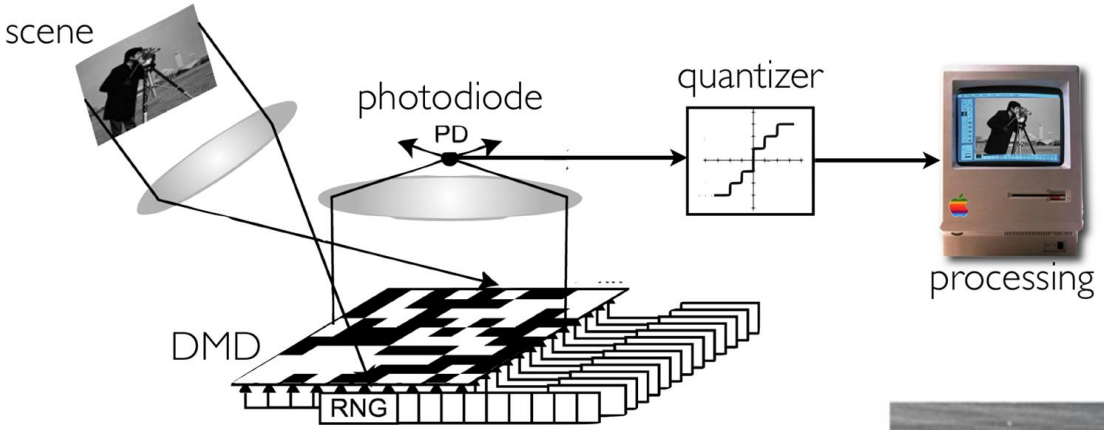
$$\arg\max_X \; p(Y|X)\,p(X)$$

Problem Specific

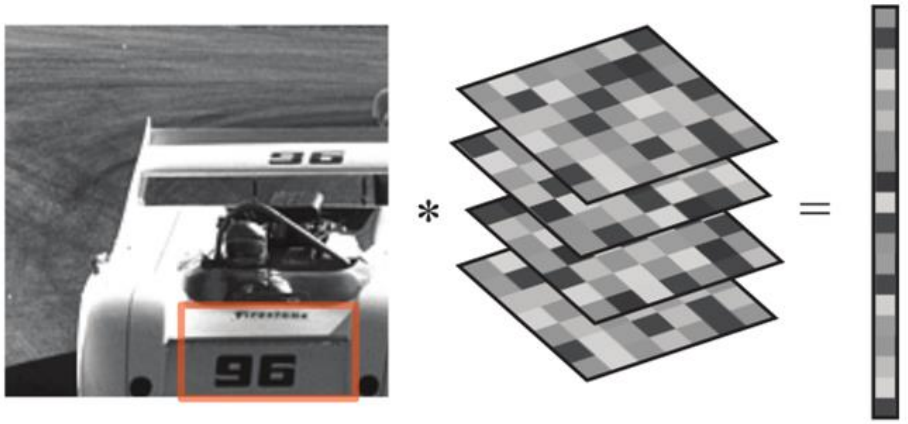
Learned by generative model

One **deep prior** for all problems

Single Pixel Camera

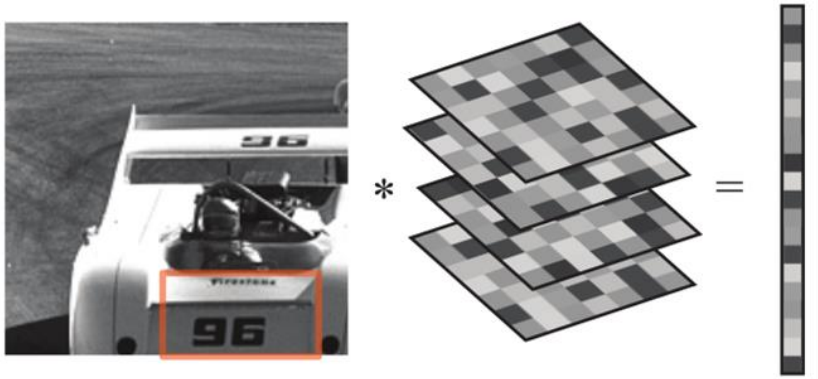


SPC by DSP lab, Rice university



Measurement acquisition simulation

RIDE for SPC reconstruction



Measurement acquisition

$$\hat{\mathrm{x}} = arg\max_{x} p(\mathrm{x})\ s.t.\ \mathrm{y} = \Phi\mathrm{x}$$

Inference

While k < max_iter:

- Gradient ascent

$$\hat{\mathrm{x}}_k = \mathrm{x}_{k\text{-}1} + \eta \nabla_{\mathrm{x}_{k-1}} p(\mathrm{x})$$

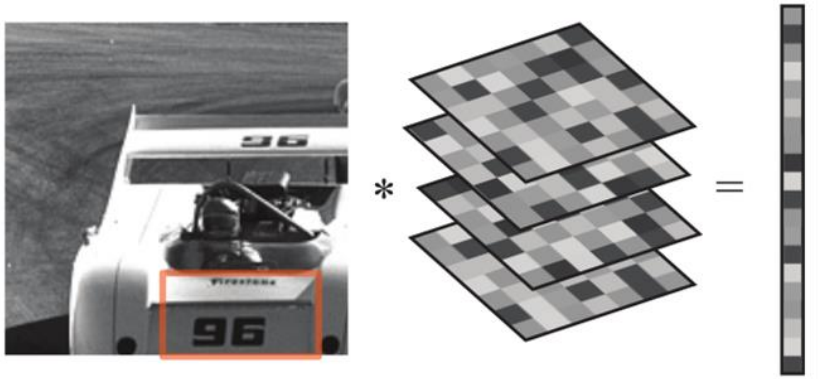
- Project the solution

$$\mathrm{x}_k = \hat{\mathrm{x}}_k - \Phi^T(\Phi\Phi^T)^{-1}(\Phi\hat{\mathrm{x}}_k - \mathrm{y})$$

Compressive Image Recovery Using Recurrent Generative Model, ICIP 2017

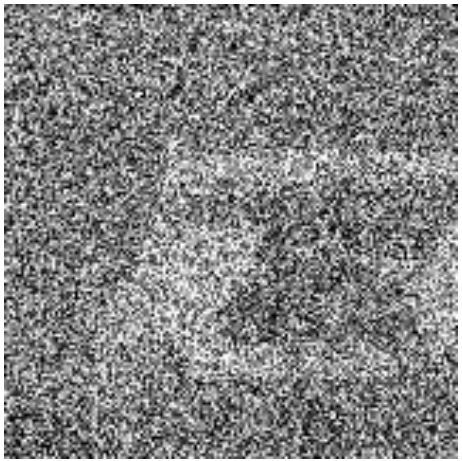
Akshat Dave, Anil Kumar Vadathya, Kaushik Mitra

RIDE for SPC reconstruction



Measurement acquisition

$$\hat{\mathrm{x}} = arg \max_{x} p(\mathrm{x})\ s.t.\ \mathrm{y} = \Phi \mathrm{x}$$



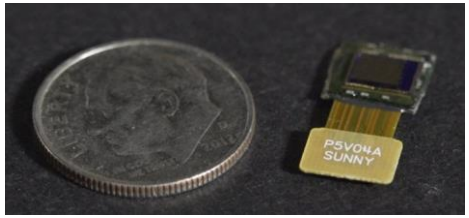
Reconstruction at 30% measuremets as the iterations progress

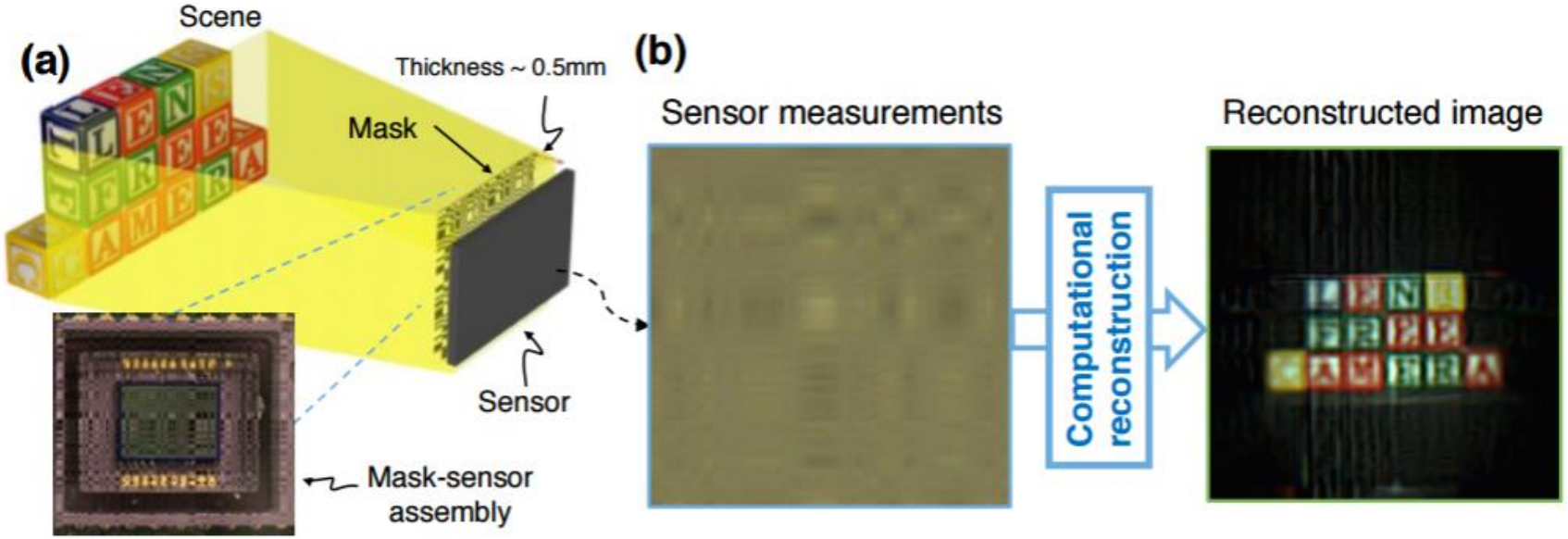
Compressive Image Recovery Using Recurrent Generative Model, ICIP 2017

Akshat Dave, Anil Kumar Vadathya, Kaushik Mitra

FlatCam - lens less imaging framework

Thin, flexible and cheap





Boominathan et. al., 2016

FlatCam: results

Original

BM3D

11.82 dB,
0.199

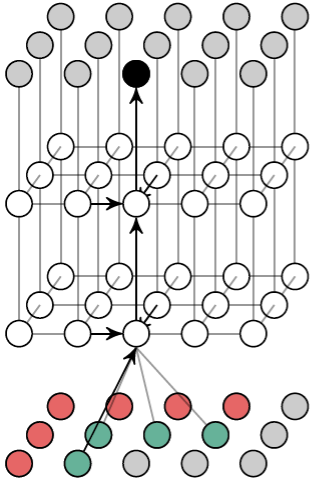
Ours

28.61 dB,
0.325

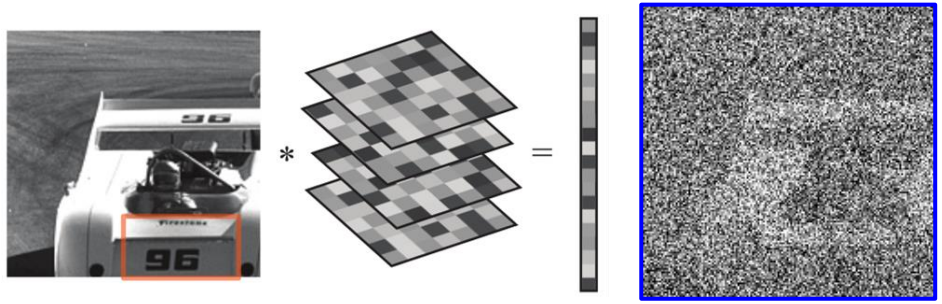
TVAL

11.79 dB,
0.226

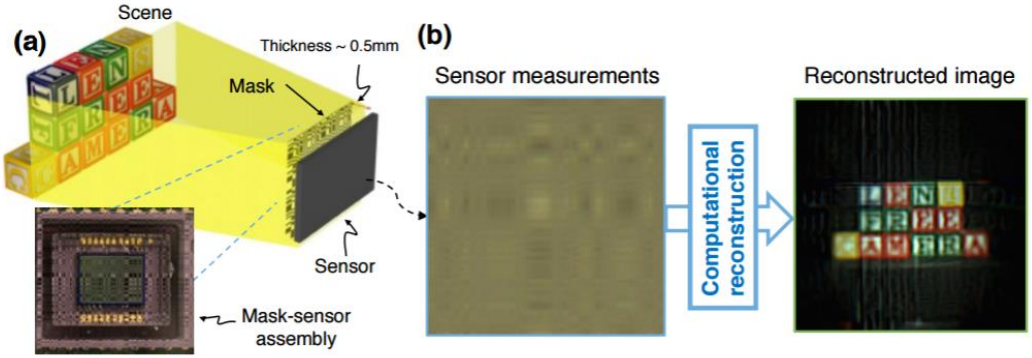
Summary



Autoregressive models: RIDE, PixelCNN/RNN, PixelCNN++, Conditional PixelCNN



Single Pixel Camera - reconstructions



FlatCam - reconstructions