

# **Large Scale Deep Learning and Self-Supervision in Vision and NLP**

Thomas Breuel

NVIDIA

# INTRODUCTION

# Topics

- using large amounts of unlabeled training data
- self-supervised training
- LSTM, transformers, ViT
- large language models
- diffusion models

Application: OCR (text recognition, layout recognition, information extraction)

# Different Kinds of Learning

- supervised learning
  - inputs and outputs are given
- unsupervised learning
  - only inputs are given
- semi-supervised learning
  - combine supervised and unsupervised data
- self-supervised learning
  - algorithm derives a supervised problem from unsupervised data

# Different Kinds of Learning

- transfer learning
  - use a model trained on one problem to solve another problem
- active learning
  - the learning algorithm requests transcribed data from an oracle
- metric learning
  - learn a distance measure that helps with clustering / classification
- representation learning
  - transform input vectors into another space that, removes noise, expresses invariances, places "similar" samples closer together

# Different Kinds of Models

- discriminative vs generative
- linear models
- non-linear models
  - parametric probabilistic
  - support vector machines
  - deep learning
    - convolutional
    - recurrent
    - transformer

# Overview

Motivation:

- OCR example

Classical Techniques:

- statistical basis of machine learning (mostly review)
- linear methods (mostly review?)

Deep Learning:

- language modeling and sequence learning
- self-supervised learning for images
- generative modeling (VAE, flows, GANs)

# Background

I'm assuming you have

- done some supervised deep learning, e.g., trained MNIST, ImageNet, etc.
- understand the basics of SGD, loss functions, etc.

# Focus

We will focus on the unsupervised / self-supervised aspects of methods and papers we will be discussing, discussing other aspects of papers only as needed.

# OPTICAL CHARACTER RECOGNITION

# Motivating Example: OCR

Optical Character Recognition:

- image → text
- closely related to object recognition, autonomous driving, speech recognition, ...
- much better statistical understanding of...
  - class structure
  - noise
  - ground truth
  - syntactic structure / language modeling

# Applications

- surprisingly, a hot topic!
- wanted: information extraction from unstructured documents
- NLP technology didn't use to be up to it
- with new LLMs, that has changed dramatically

# Motivating Example: OCR

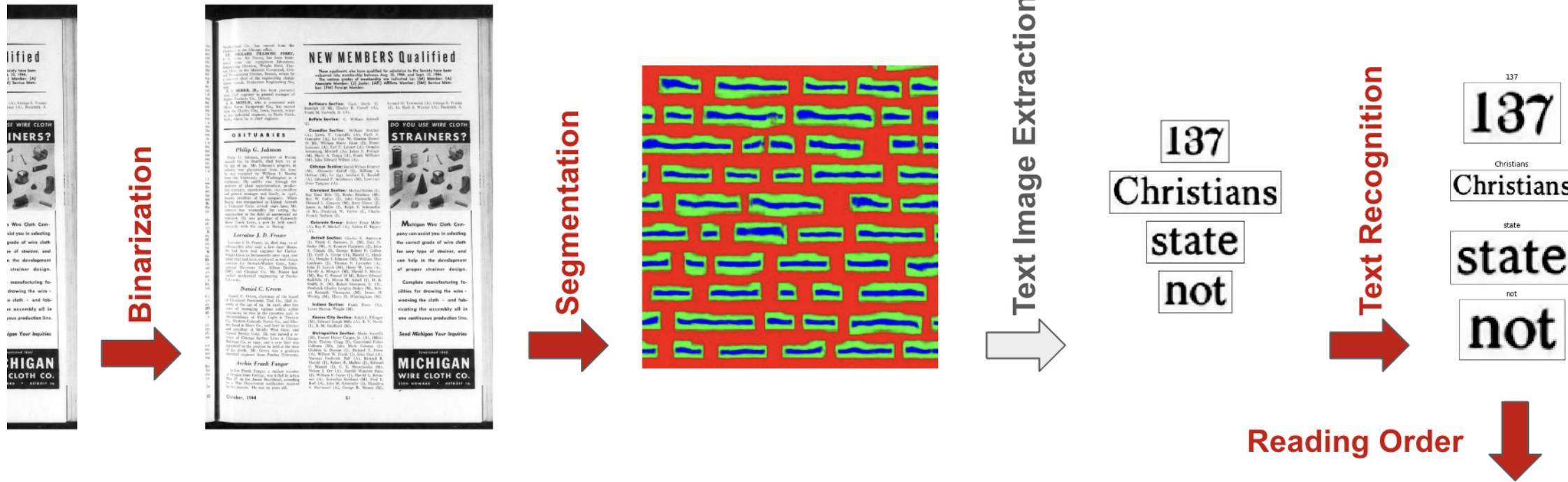


```
xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
<title></title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
<meta name="ocr-system" content="tesseract 4.1.1-cv2-25-g9707" />
<meta name="ocr-capabilities" content="ocr_page ocr_carea ocr_par ocr_line ocrx_word ocrp_xconf"/>
</head>
<body>
<div classe="ocr_page" id="page_1" title="image.sample.jpg"; bbox 0 0 2130 3433; ppageno 0>
<div classe="ocr_carea" id="block_1_1" title="bbox 115 224 1884 284">
<p><span classe="ocr_par" id="par_1_1" lang="eng" title="bbox 115 224 1884 284">
<span classe="ocr_line" id="line_1_1" title="bbox 115 224 1884 284"; baseline -0.001 -11; x_size 51.400451; x_descenders 5.400458; x_ascenders 10">
<span classe="ocrx_word" id="word_1_1" title="bbox 115 224 220 284"; x_conf 75'-1654'>
<span classe="ocrx_word" id="word_1_2" title="bbox 228 224 328 284"; x_conf 93'-Prerogative</span>
<span classe="ocrx_word" id="word_1_3" title="bbox 336 224 436 284"; x_conf 71'-Court</span>
<span classe="ocrx_word" id="word_1_4" title="bbox 444 224 544 284"; x_conf 96'-of</span>
<span classe="ocrx_word" id="word_1_5" title="bbox 552 224 652 284"; x_conf 73'-Canterbury.</span>
<span classe="ocrx_word" id="word_1_6" title="bbox 660 224 760 284"; x_conf 87'-273</span>
</span>
</p>
</div>
<div classe="ocr_carea" id="block_1_2" title="bbox 118 292 1792 3433">
<p><span classe="ocr_par" id="par_1_2" lang="eng" title="bbox 118 292 1792 3433">
<span classe="ocr_line" id="line_1_2" title="bbox 118 292 1792 3433"; baseline 0 3132; x_size 28; x_descenders 5">
<span classe="ocrx_word" id="word_1_7" title="bbox 118 292 1792 3433"; x_conf 95'>
</span>
</p>
</div>
<div classe="ocr_carea" id="block_1_3" title="bbox 116 350 178 3433">
<p><span classe="ocrx_word" id="word_1_8" title="bbox 116 350 178 3433"; x_conf 90'-SLADEN,</span>
<span classe="ocrx_word" id="word_1_9" title="bbox 186 350 246 3433"; x_conf 94'-Janes,</span>
<span classe="ocrx_word" id="word_1_10" title="bbox 254 350 314 3433"; x_conf 93'-of</span>
<span classe="ocrx_word" id="word_1_11" title="bbox 322 350 382 3433"; x_conf 91'-g,</span>
<span classe="ocrx_word" id="word_1_12" title="bbox 390 350 450 3433"; x_conf 96'-st,</span>
<span classe="ocrx_word" id="word_1_13" title="bbox 458 350 518 3433"; x_conf 96'-Margaret,</span>
<span classe="ocrx_word" id="word_1_14" title="bbox 526 350 586 3433"; x_conf 96'-near,</span>
<span classe="ocrx_word" id="word_1_15" title="bbox 594 350 654 3433"; x_conf 96'-Rochester,</span>
<span classe="ocrx_word" id="word_1_16" title="bbox 662 350 722 3433"; x_conf 96'-Kent,</span>
<span classe="ocrx_word" id="word_1_17" title="bbox 730 350 790 3433"; x_conf 96'-July,</span>
<span classe="ocrx_word" id="word_1_18" title="bbox 798 350 858 3433"; x_conf 96'-27,</span>
<span classe="ocrx_word" id="word_1_19" title="bbox 866 350 926 3433"; x_conf 96'-1653.</span>
</span>
</p>
</div>
</div>
</body>

```

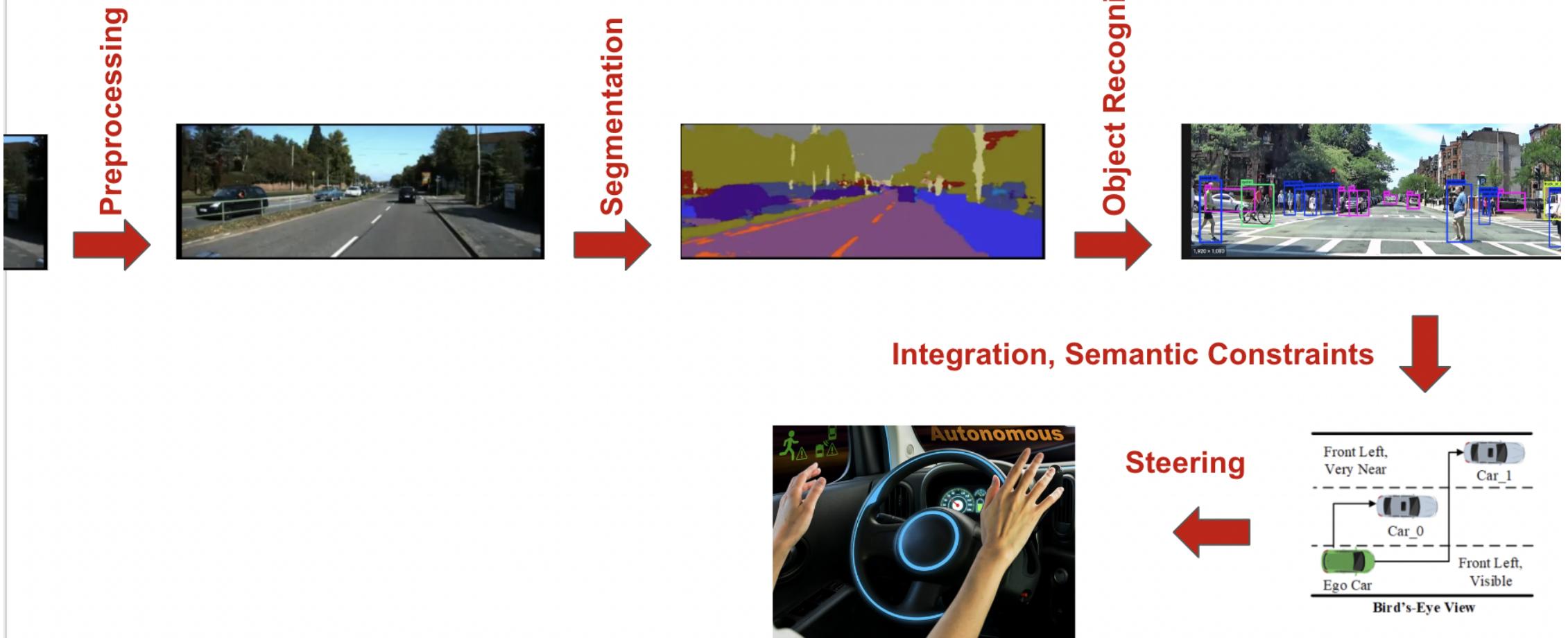
image → text, markup

# Motivating Example: OCR



1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not in their exact original form, reproduced in their exact original form,

# OCR vs AV Pipeline



# Motivating Example: OCR

We need to train multiple models:

- preprocessing
- segmentation
- recognition
- syntactic model (language model)
- geometric model (reading order model)

This is common to many computer vision tasks using deep learning.

# OCR and Semi-Supervised Learning

Available data:

- thousands of scanned pages with manual segmentation and text
- tens of thousands of scanned pages with approximate text
- millions of scanned pages without text
- large amounts of text without document images
- the ability to generate new texts and printed documents

Typical unsupervised / semi-supervised learning problem.

# First Step: Supervised Training

Train supervised models on manually labeled / transcribed data:

- segmentation model
- image → text model

Yields good performance *on data similar to training data.*

Unlabeled training data primarily helps with generalization to new datasets.

# Using Millions of Untranscribed Pages

Idea:

- Use our or other OCR systems to transcribe those pages and use the output as training data ("pseudolabel")

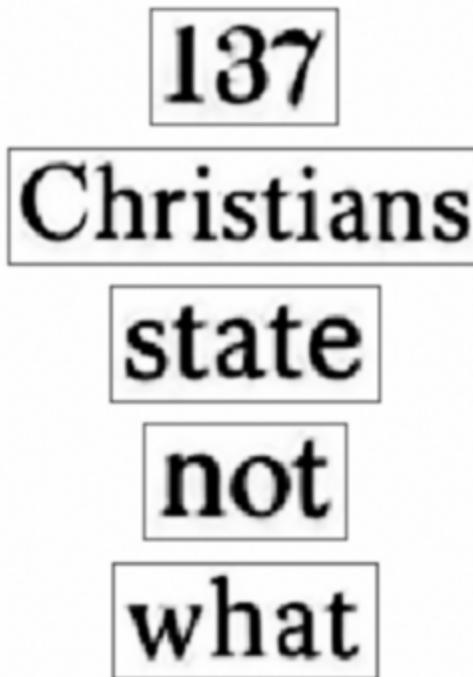
Questions:

- How can using a worse OCR system work for training a better OCR system?
- Is there anything we can do to improve this?

# **SEMI-SUPERVISED TEXT RECOGNITION**

# Example: Word Recognition Problem

Let's focus just on recognizing words for the following examples (forget about the rest of the OCR system):



137  
Christians  
state  
not  
what

# Using Millions of Untranscribed Pages

Idea:

- run the existing OCR system, giving word images and corresponding text
- construct a new training set by...
  - rejecting any word that the OCR system has a low confidence in
  - rejecting any word that is not found in the dictionary
- this way, we obtain a new training set that contains "mostly good" training samples
- iterate this multiple times

# Why does this work?

Network estimates  $P(c|x)$  ( $x$ : image,  $c$ : class)

For OCR, we know:

- true  $P(c|x)$  is approximately 0 or 1 (no ambiguities)
- any uncertainty in classifier output is due to mislabeled training data
- e.g. if 20% of training data mislabeled:  $\tilde{P}(c|x) = 0.8$  for true class  $c$
- if we use pseudolabel  $\arg \max_c \tilde{P}(c|x)$ , many accidentally mislabeled training labels will actually be correctly labeled
- as a result, the posterior probability will be estimated higher on the next training round and the model improves

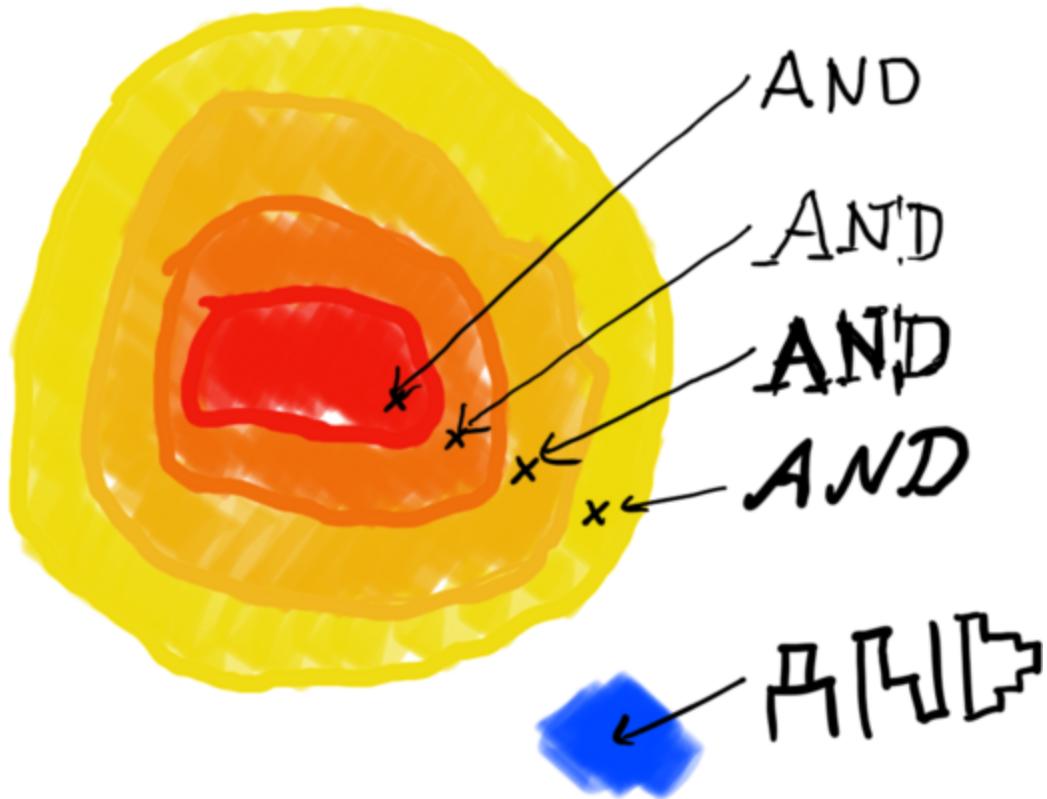
# EM Algorithms

- latent variables in semi-supervised OCR
  - labels for unlabeled portion of training set
  - outlier status for unlabeled portion of training set
- EM algorithms recover latent variables by...
  - "making a best guess" given the current model
  - retraining the model as if that guess is correct

# Unsupervised Learning

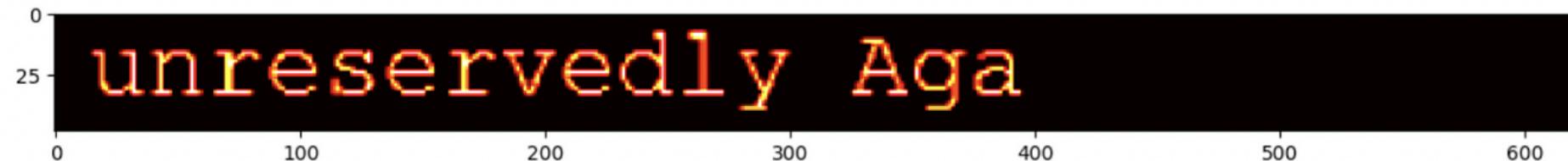
- identify the *latent variables* that are being recovered
- identify the *prior assumptions / inductive biases* in the model
- identify the *EM algorithm implementation*

# Iterated Recognition / Dataset Construction

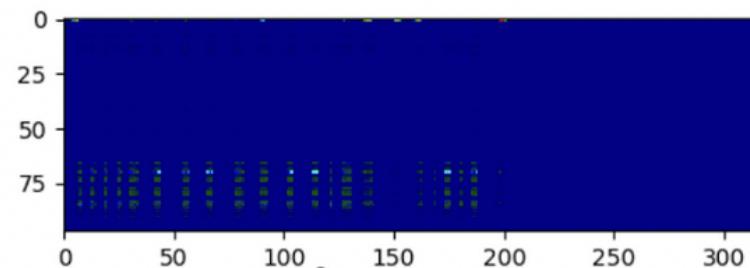


# WEAKLY SUPERVISED TEXT RECOGNITION

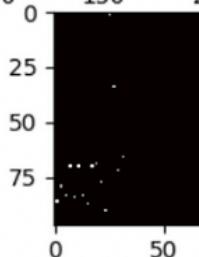
# EM Training with Language Models (aka CTC)



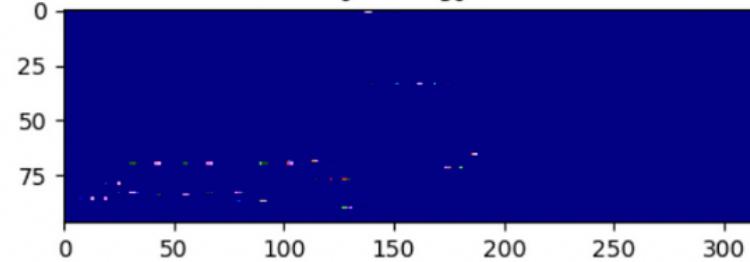
posterior probability at each pixel location



one-hot ground truth



CTC-aligned ground truth



## EM Training with LSTM/Conv + CTC

- ideally, the LSTM/Conv model outputs the correct class only where the character occurs
- outputs  $\epsilon$  (no character) everywhere else
- our transcript does not contain the character positions
- CTC performs *alignment* between classifier output and transcript
- CTC estimates the most likely positions of characters given the current model



# EM Training without Any Transcripts

Normal EM-Training:

- image + transcript, EM-training recovers alignment

Language Model-Based EM-Training:

- image + language model, EM-training recovers text + alignment

# EM Training, Information Theory

There always has to be *some* source of information:

- unigram perplexity: 1000 ← what the classifier outputs
- bigram perplexity: 200
- trigram perplexity: 100 ← what the language model imposes
- full transcript perplexity: 1 ← fully supervised training

Additionally: several bits per character for character location.

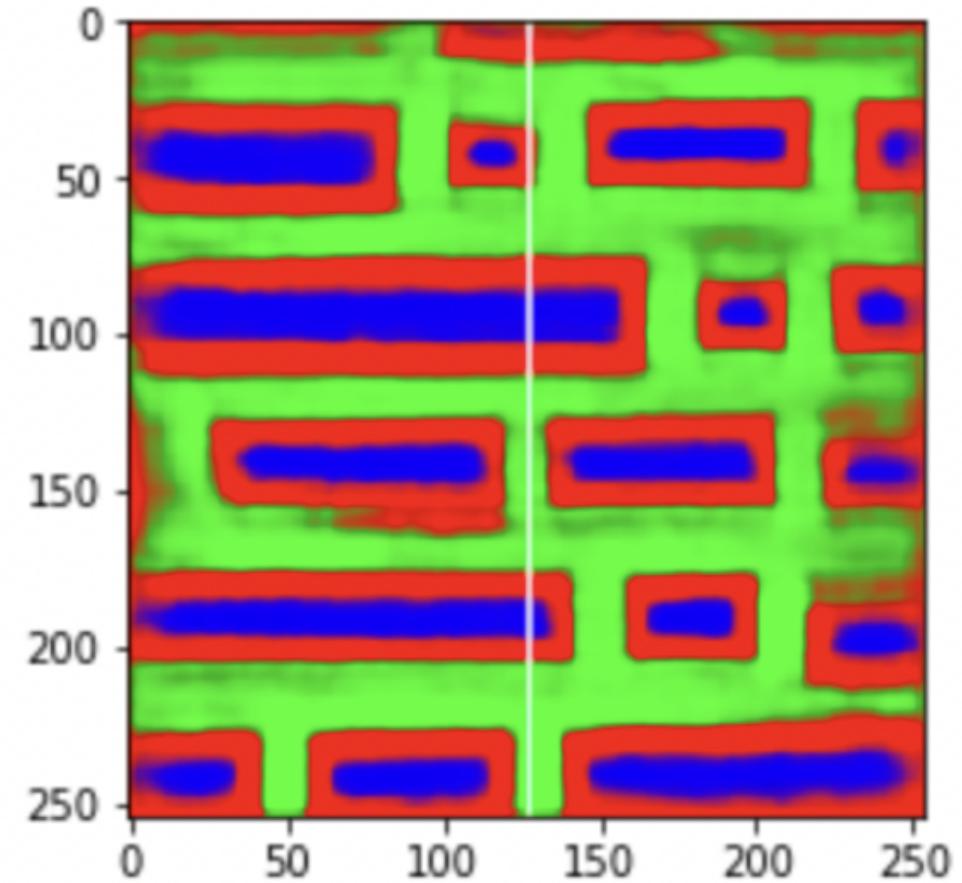
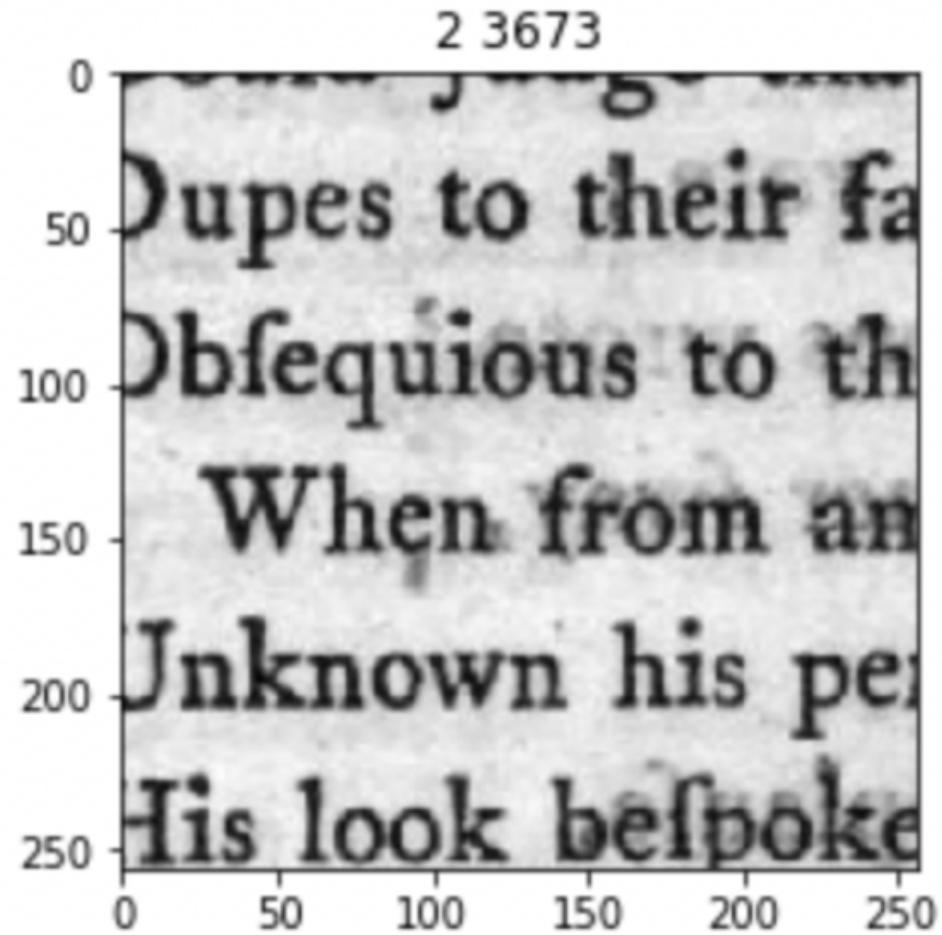
# Different Forms of Weak-/Unsupervised Learning

We have seen three different forms of weak/unsupervised training for text → image training:

- using image + transcript, lacking just the alignment/segmentation
- using image + language model only
- using just pseudolabels from weak classifier

# **SELF-SUPERVISED PAGE SEGMENTATION**

# Self-Supervision of Page Segmentation



# Self-Supervision of Page Segmentation

Assume we start with errorful segmenter from small labeled training data.

Two kinds of errors:

- segmenter returns word for non-words
- segmenter misses actual words

Approach:

- validate each returned word using OCR (no OCR = not a word)
- mark everything that is not identified as a word as "don't know" and exclude from training

# Self-Supervision

Approaches to self-supervision usually incorporate prior knowledge and require some manual design:

- text recognizer
  - reject non-words from "soft labels"
  - use language models as part of EM training
- page segmentation
  - validate segmentation via OCR
  - introduce "don't know" mask during training

# Self-Supervision

Prior knowledge about the task is required to choose meaningful self-supervision tasks.

- word recognition: pseudolabels, language-model based rejection
- segmentation: verification via OCR, introduction of "don't care" regions
- object recognition, natural image segmentation: *later*

# ACTIVE LEARNING

# Active Learning

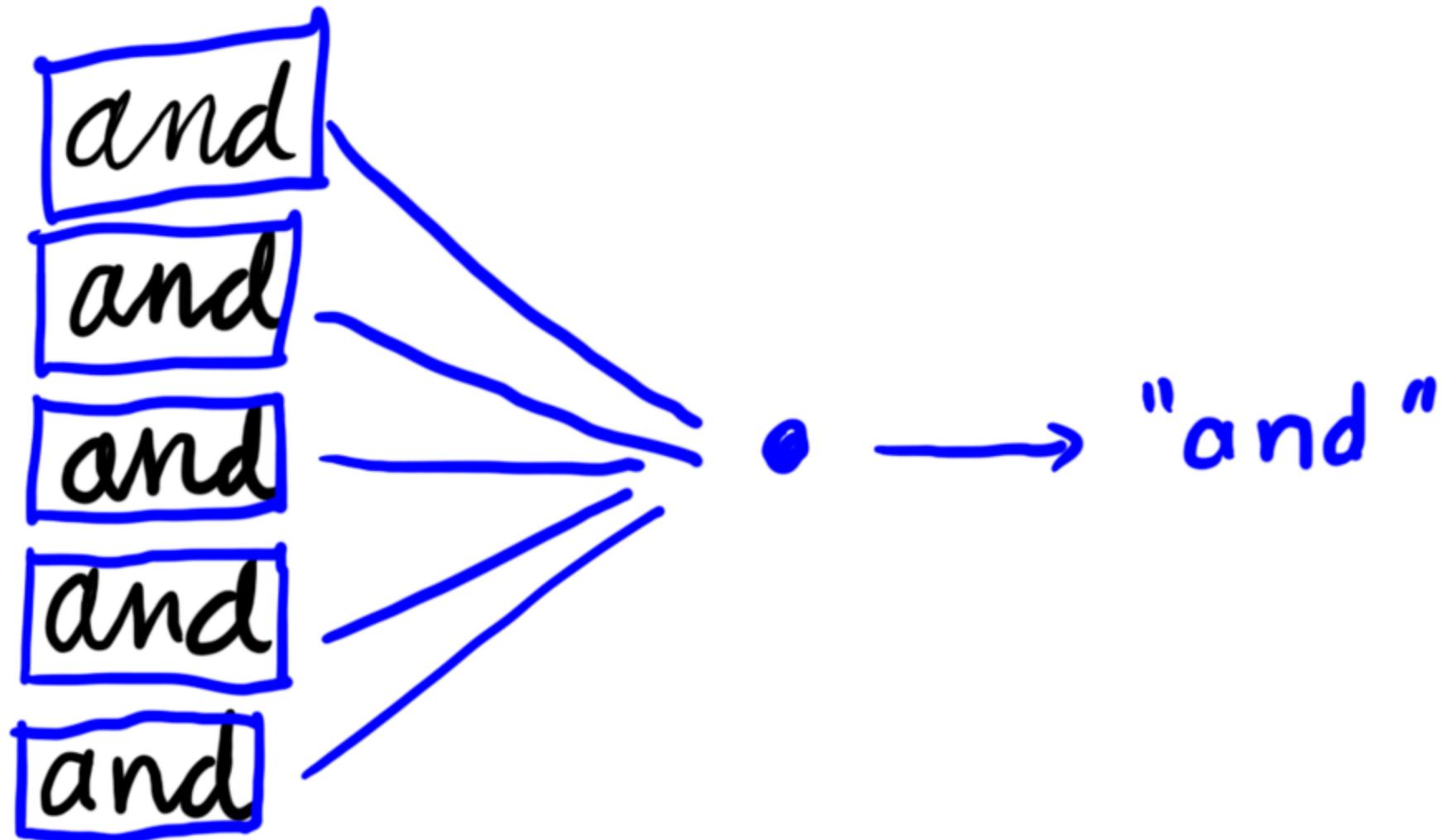
Request Help from Oracle:

- run OCR as above and send low confidence outputs to manual transcribers
  - output near decision boundaries
  - output that can maximally help the classifier improve

Clustering:

- perform clustering on character or word images and manually transcribe each cluster

# Transcription with Clustering



Question: what similarity measure do we use for clustering?

# OCR by Solving a Cryptogram

የእውቅና የአዲስ አበባ ቤት ስራ ተደርጓል እንደሆነ ተደርጓል የሚያሳይ

- cluster characters by shape
  - use frequency and patterns to infer character identity
    - e.g.: first word is likely "THE"
      - letter frequencies
      - word frequencies
    - single letter word is likely "A"
  - explicit form of many EM-based recognition algorithms

# OCR by Solving a Cryptogram

THE CAT SAT ON A WALL AND LOOKED AT THE SUN.  
A DOG SAT ON THE STREET AND CHEWED A BONE.

# DATA SOURCE MODELS

# Hierarchical Bayes



- clustering is based on a simple hierarchical Bayesian source model
- latent variables can be recovered by EM or Bayesian methods
- simple clustering assumes  $\Sigma = 1$
- both  $c$  and  $\mu$  are latent; need oracle to recover actual class labels

# Channel View

$$c \in \mathbb{Z}_k \xrightarrow{\text{render}} \xi \xrightarrow{\text{noisy channel}} x \xrightarrow{\text{classifier}} P(c|x)$$

OCR: render = digital typeset; noisy channel: printing + scanning

Vision: render = choice of pose ; noisy channel: lighting, camera

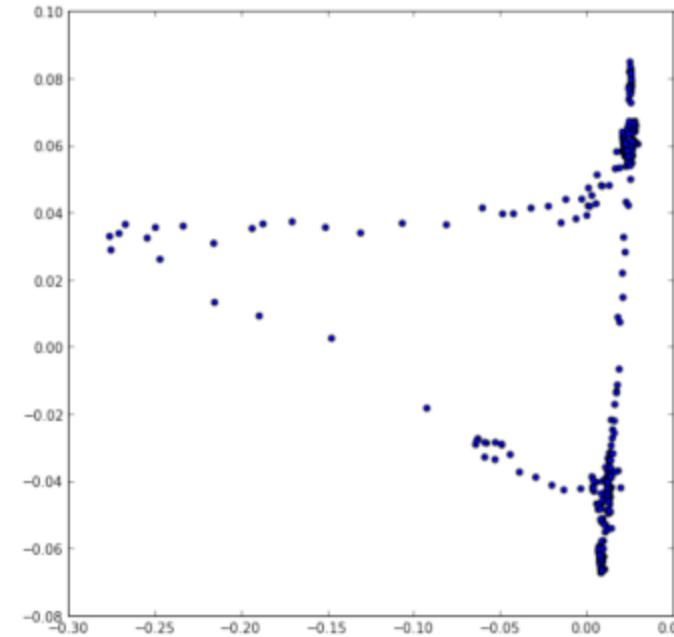
Rendering can involve transformation parameters, giving rise to *view manifolds*.

# Real View Manifold



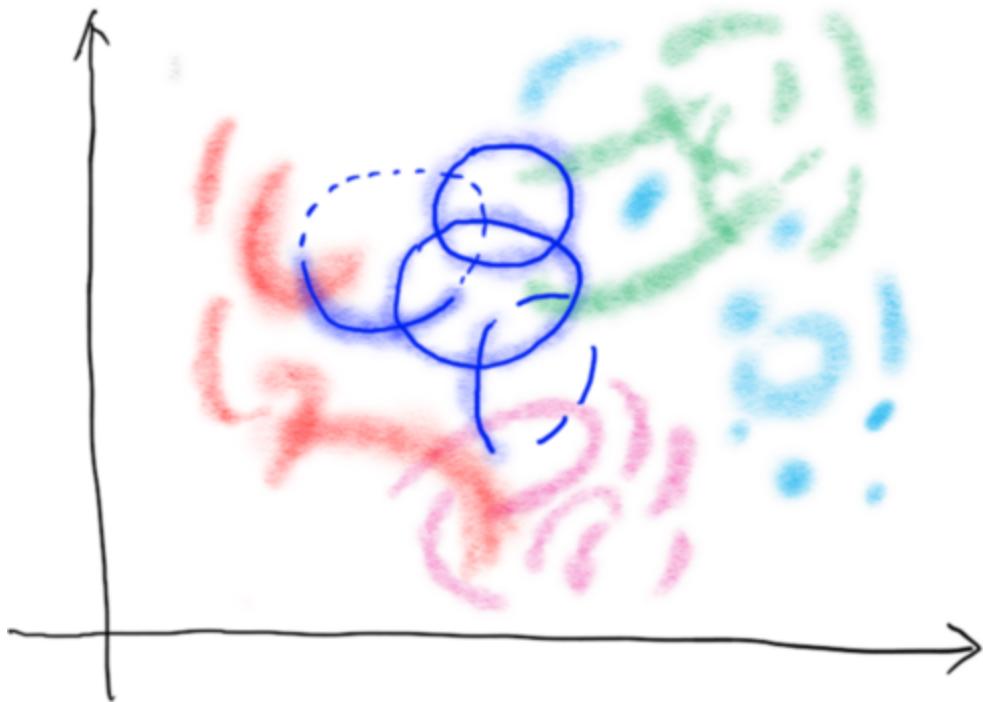
5 second motion sequence of object rotating back and forth

```
from sklearn.decomposition import RandomizedPCA  
pca = RandomizedPCA(20)  
lo = pca.fit_transform(data)  
mds = manifold.LocallyLinearEmbedding()  
vl = mds.fit_transform(lo)  
scatter(vl[:,0],vl[:,1])
```



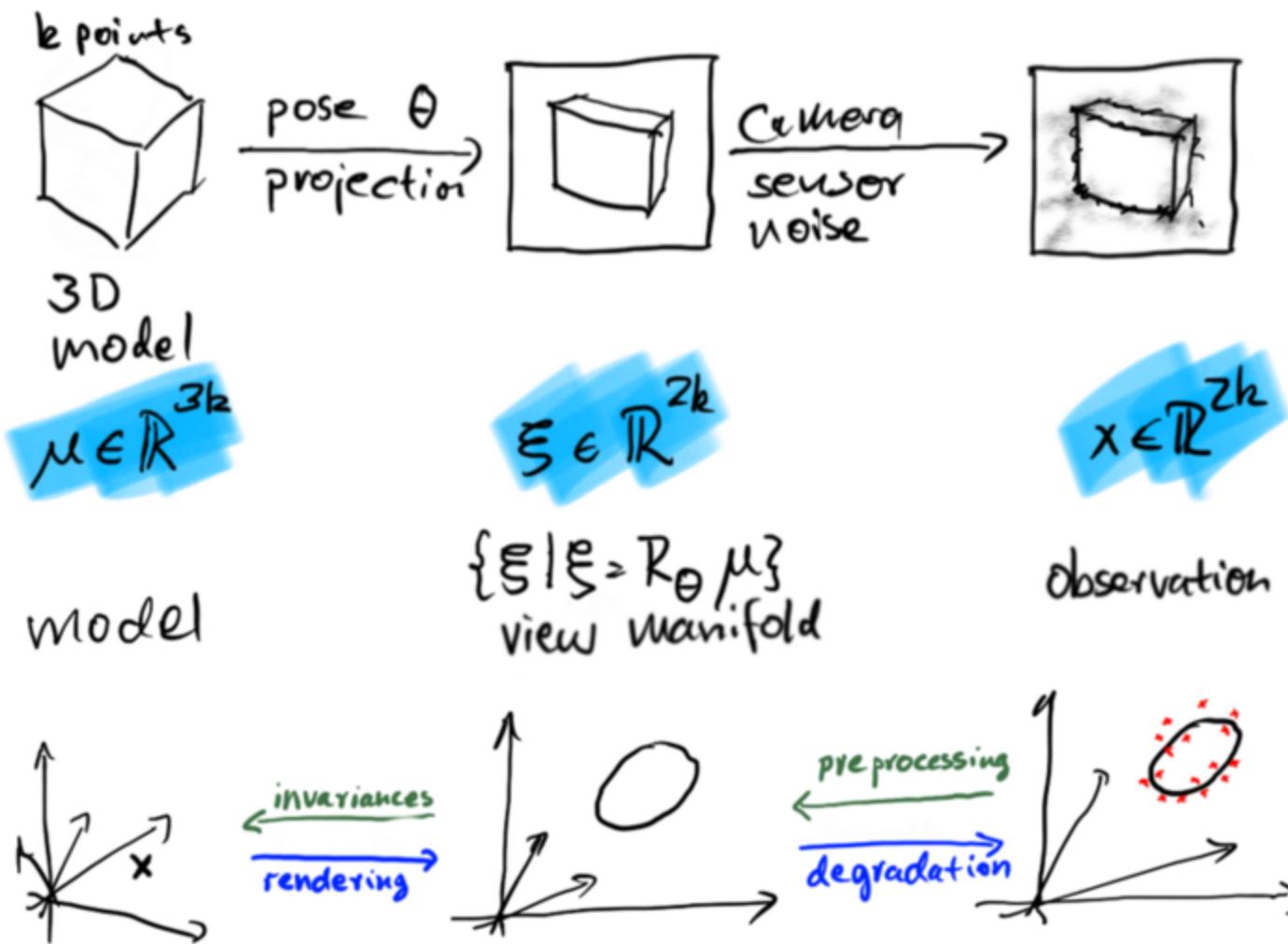
Dimensionality reduction in pixel space:  $\mathbb{R}^{256} \rightarrow \mathbb{R}^2$ . Video: [rotating.mp4](#)

# Manifolds Learning



- think of real data as clouds surrounding...
- union of manifolds with boundary (only shown for blue class)
- "manifold learning" = union-of-manifold-s-with-boundary learning

# Model, View Manifold, and Noise



# Unsupervised Learning

Given or Learned:

- rendering
- degradation

Constructed or Learned (inverse problems):

- invariances
- preprocessing / image cleanup

CycleGAN can learn both directions simultaneously with no supervision.

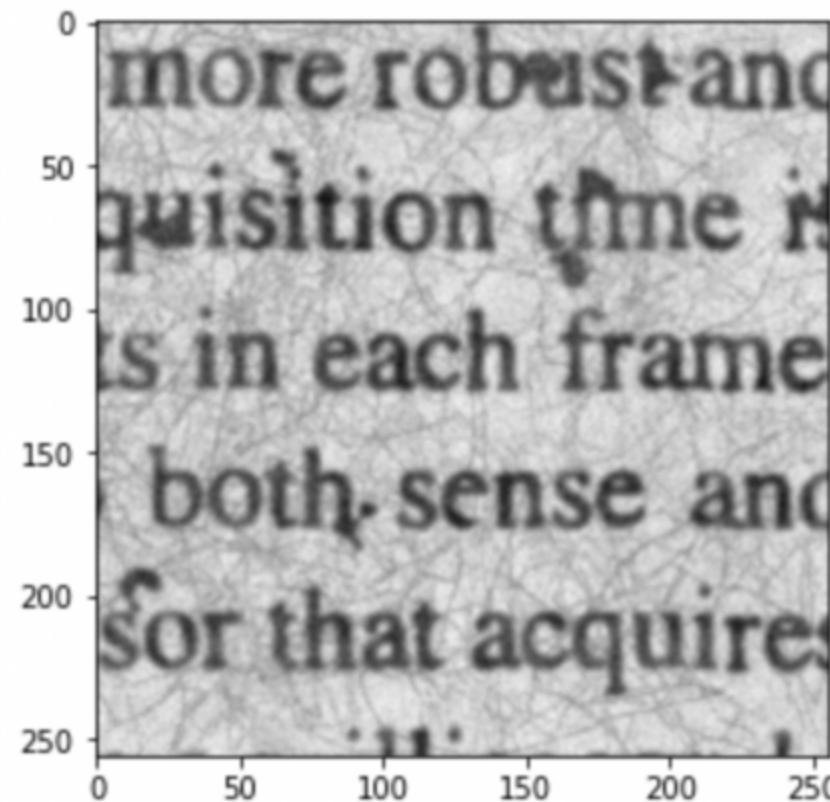
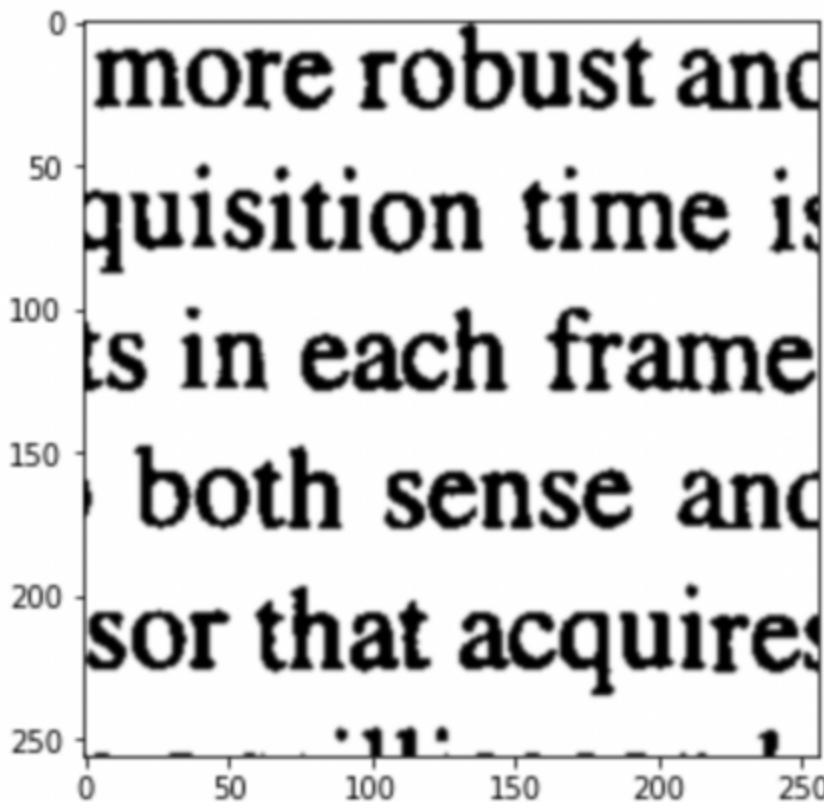
# **TRAINING DATA GENERATION / DEGRADATION**

# Artificial Data Generation

This is the "forward path" in the channel view of recognition:

- digital typesetting = perfect "artificial" document generation
- take any text, generate pages of perfect text
- for OCR, we need "degraded images"
  - printing, scanning, photographing, photocopying, ...
- physical processes for document image degradataion are well known

# Artificial Data Generation



# Document Image Restoration

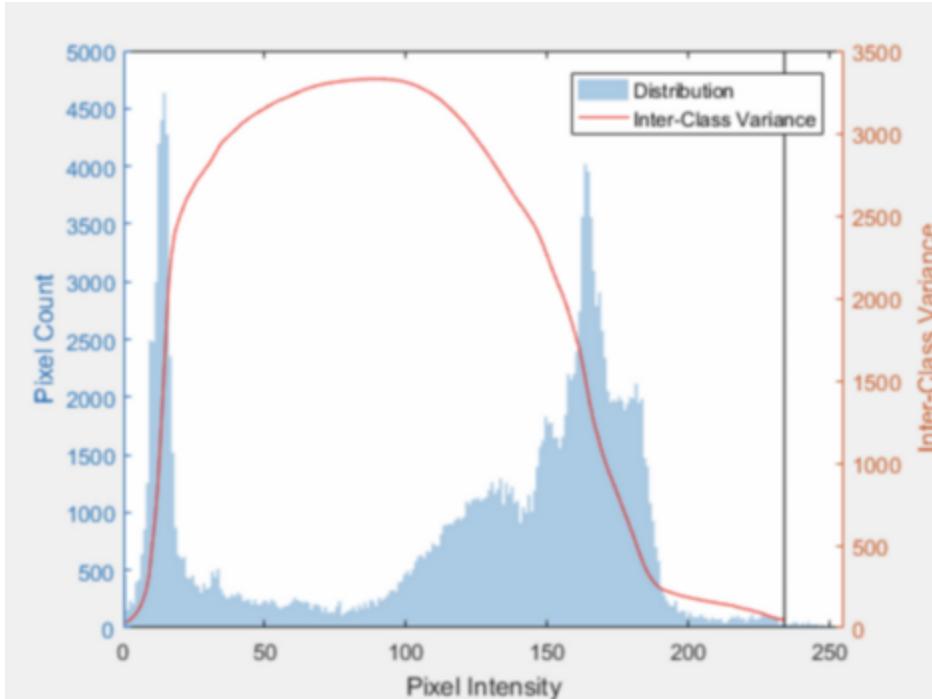
- recognition is easier if documents are not degraded
- can we restore the "clean" image via unsupervised learning?

AV/object recognition: image translation prior to recognition

# Classical Document Image Restoration

- binarization
  - optimal thresholding
  - optimal linear filtering (deconvolution, etc.)
  - clustering
  - performance-based dynamic thresholding
- deep learning
  - supervised restoration (LSTM, pix2pix)
  - unsupervised restoration via CycleGAN

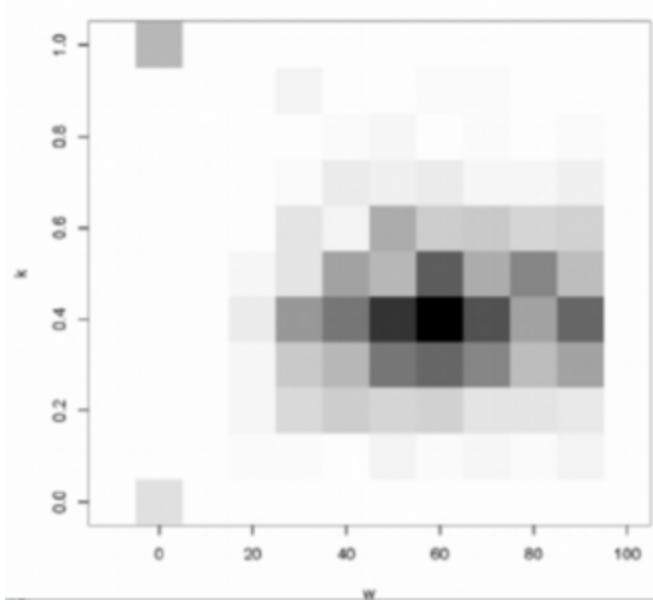
# Otsu's Method: "Optimal" Thresholding



Otsu's method:

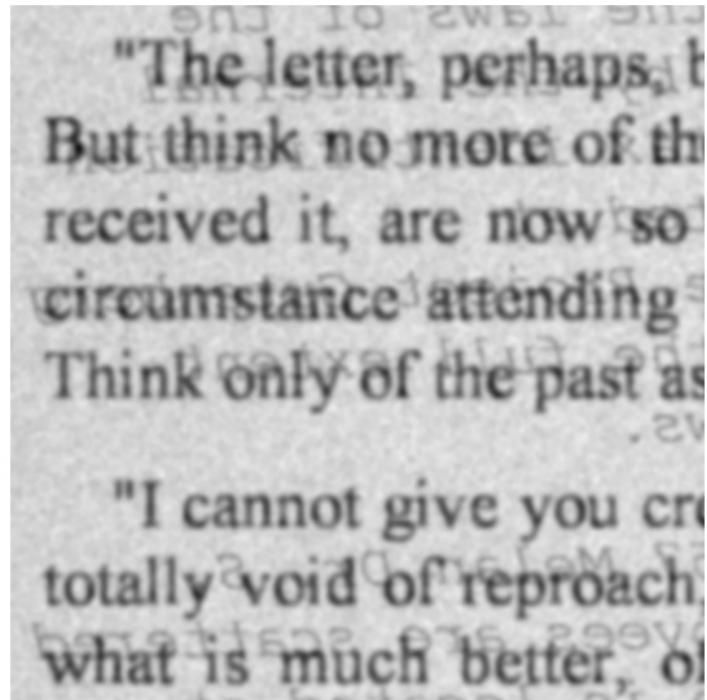
- manually constructed inverse degradation model based on prior knowledge/assumptions
- assume some kind of mixture model of image pixel generation
- maximize inter-class variance

# Performance-Based Thresholding



- run thresholding with many parameters
- pick the parameters that yield the best OCR output
- no transcript, so use proxy
  - use statistics/classifier, or #words in dictionary
  - closely related to GAN methods
- assumption/prior: local thresholding = good way of inverting degradation model

# Binarization by Supervised Deep Learning



(a)

"The letter, perhaps, it  
But think no more of the  
received it, are now so  
circumstance attending  
Think only of the past as

"I cannot give you cre  
totally void of reproach,  
what is much better, ol

(b)

"The letter, perhaps, it  
But think no more of the  
received it, are now so  
circumstance attending  
Think only of the past as

"I cannot give you cre  
totally void of reproach,  
what is much better, ol

(c)

(a) original, (b) Sauvola, (c) LSTM-based binarization

# Self-Supervised Training for Binarization

1. generate clean images, degrade with degradation model
  - takes advantage of prior knowledge of degradataion models
2. take degraded images, use performance-based thresholding
  - takes advantage of knowledge of statistical properties of output

# CycleGAN for Document Preprocessing

CycleGAN replaces all those components with trainable networks:

- clean image → degraded image
- degraded image → clean image
- clean image detector
- degraded image detector

These correspond to the forward and backward arrows in our channel model.

CycleGAN can be trained end-to-end without any labeled data or (significant) prior assumptions.

# CycleGAN for Document Preprocessing

EMOIR OF THE AUTHOR.

his thought, that nothing is i  
; of Christ to engage in, i  
effectually promote the ki  
laker. Perhaps it is not p  
d the world will hardly belie  
een taken in composing th  
what care I have endeav

 $d_A$ 

$$g_{AB} \rightarrow$$

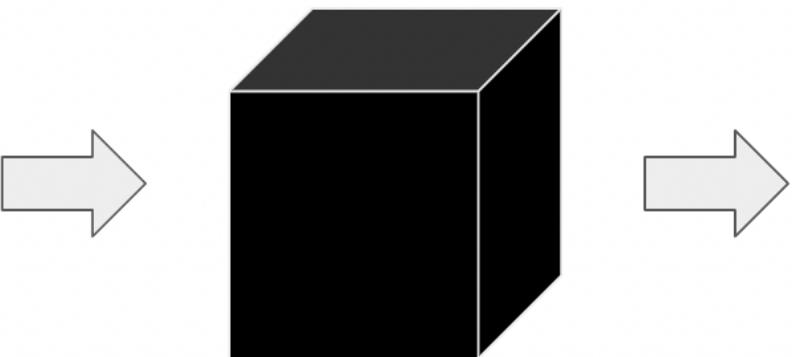
$$\leftarrow g_{BA}$$

EMOIR OF THE AUTHOR.

his thought, that nothing is i  
; of Christ to engage in, i  
effectually promote the ki  
laker. Perhaps it is not p  
d the world will hardly belie  
een taken in composing th  
what care I have endeav

 $d_B$

# End-to-End OCR



Weatherhead Co., has moved from the Cleveland to the Chicago office.

**LT. MILLARD FILLMORE PERRY, U. S. Army Air Forces,** has been transferred from the equipment laboratory, Engineering Division, Wright Field, Dayson, Ohio, to the Materiel Command, Central Procurement Division, Detroit, where he is assistant chief of the engineering change branch.

...

# End-to-End OCR

## Recent Developments

- transformer models permit full end-to-end training for image-to-text transcriptions
- do not need intermediate segmentation, text-line recognition

# LayTR Multicolumn Recognition

Image-to-HTML tags.

# LayTR Table Recognition

tweezed	hydroxylase	preinvasive	filices	unlabialising
annalists	implodes	lachrymation	soir	scleroprotein
yashiro	zibeth	vespering	crotchets	ascescency
hundredponny	trigonoid	aquatical	limburgito	prododication
similarize	rhizocephala	brooklike	supervalued	contrary
milligramage	cativo	yorkshireism	protension	geleem
midnightly	pedatifid	scienced	bacteriosis	germanomania

*fleahopper gallivanting  
oncoming heathenize  
breakback celebrators  
overwrestle sampler*

bayberries	cyanidin	nicotinise	stercorary	incogitant	talepyet
unlethal	snuffle	poltroonery	scimiterpod	abasedness	boomers
sediments	mulctatory	spindlewise	timocratical	renunciatory	standardizer

midianitish | elastomeric  
rodding | polymelian  
cottonwick | verbena  
smouser | gingalls  
shippound | verifiably  
melter | lurch

**Figure 3:** Four samples (a)-(d) featured in the LayTr-Tables dataset.

Image-to-HTML tags.

# Self-Supervised Training

- apply the same principles
- train an initial model using supervised data
- compute output on unlabeled data ("soft labels")
- correct the output using language models
- retrain
- possibly use auxiliary tasks for pretraining (later)

**Or has it been solved already...**



# Summary

In the OCR example, we have seen most of the major concepts of unsupervised and semi-supervised training:

- EM training (aka soft labeling)
- use of language modeling as data source
- clustering
- unsupervised preprocessing / image enhancement

- MLM trained to minimize binary cross-entropy loss on discriminator's generated labels
- Adversarial objective: Maximize discriminator's loss while minimizing MLM's loss

## ChatGPT

- Conversational AI system by Microsoft (2019)
- Pre-trained on large corpus of text data using unsupervised learning
- Possesses a vast store of knowledge and skill at reasoning on mathematical problems
- Traces the context accurately in multi-turn dialogues
- Aligns well with human values for safe use
- Supports plugin mechanism to extend capacities with existing tools or apps
- Strongest at: Multi-turn dialogues and aligning with human values for safe use

# **SEMI-SUPERVISED LEARNING IN VISION**

# Already covered...

We have already covered (in the OCR case):

- pseudo-labels
- active learning
- data augmentation and prior knowledge of invariances
- EM algorithms

# Pre-Transformer Approaches

Prior attempts to carry over principles for language models and HMMs to the image domain:

- linearize images and apply HMM or LSTM models
- apply VQ to patches and apply syntactic models to the resulting "visual words"
- corrupt images with noise and training a network to restore them
- predict color images from grayscale
- mask parts of images and predict the masked parts (like BERT)
- determine the spatial relations between patches (like entailment)

All of these yield deep learning architectures that are potentially useful for transfer learning, but never beat supervised models.

# Masked Predictions



(a) Input context

(b) Human artist

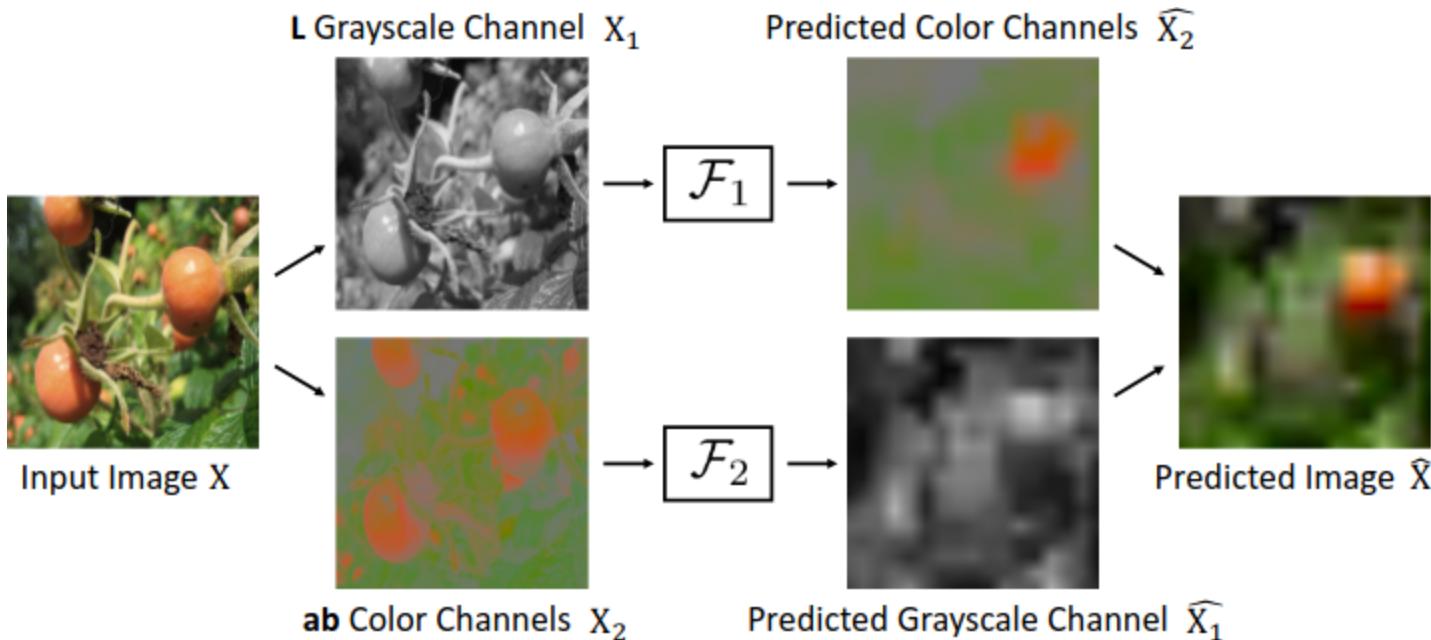


(c) Context Encoder  
( $L_2$  loss)

(d) Context Encoder  
( $L_2$  + Adversarial loss)

Pathak et al. 2016

# Split-Brain Autoencoder



Zhang et al. 2016

# Context Encoding

Example:



Question 1:



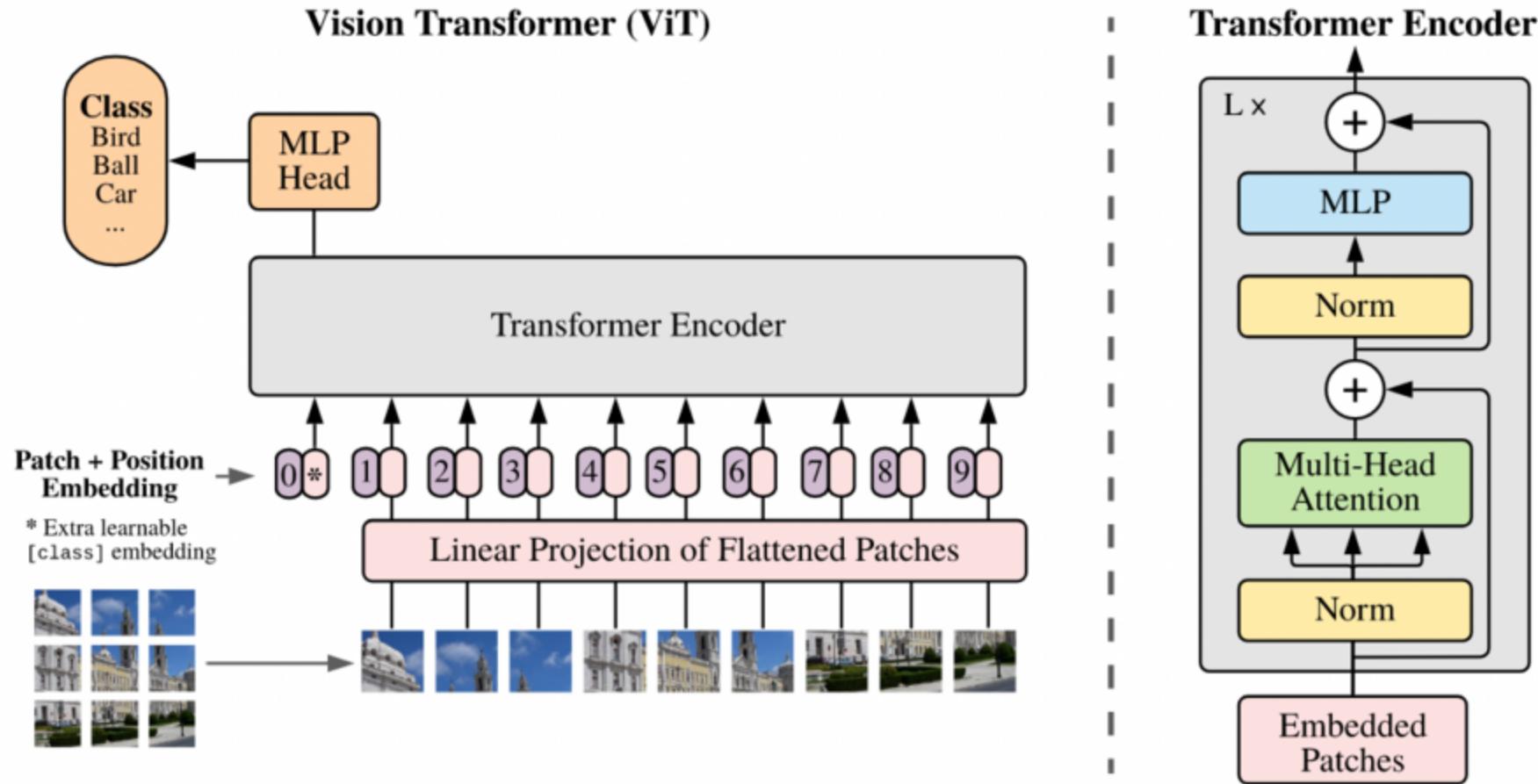
Question 2:



Doersch et al. 2016

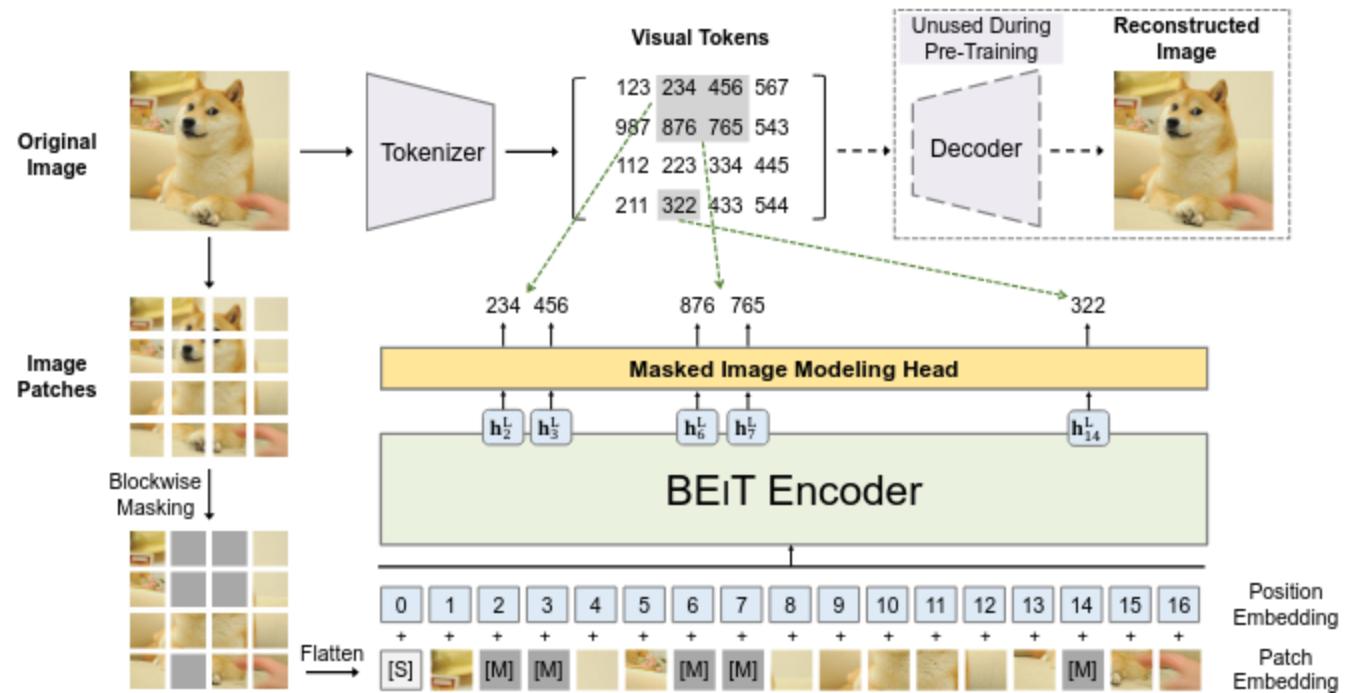
# VISION TRANSFORMERS

# Vision Transformers



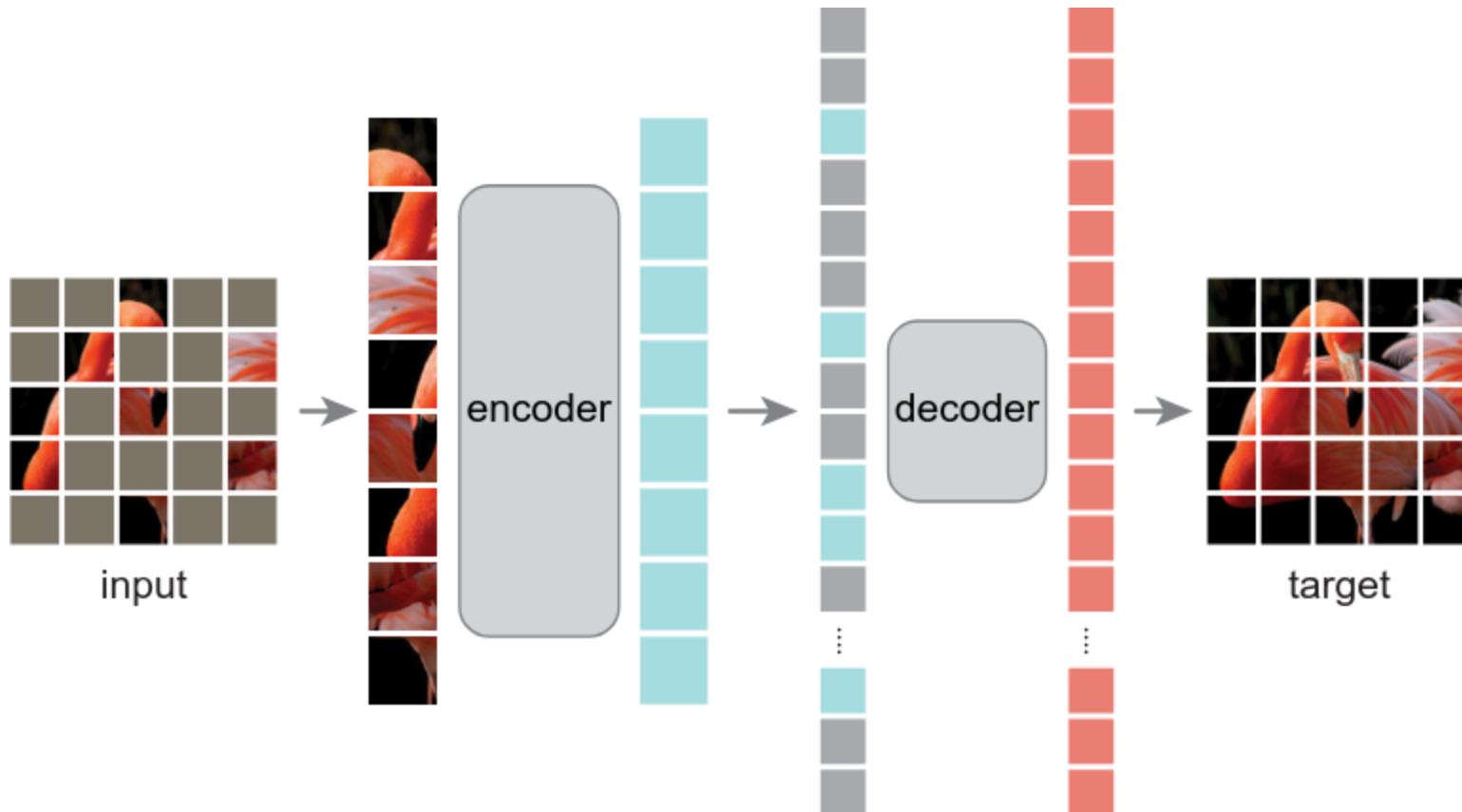
The NLP transformer architecture carries over directly to images: just change the **positional embedding**. Dosovitskiy et al. 2020, arXiv:2010.11929

# BEiT - BERT Pre-Training of Image Transformers



BERT-like pretraining carries over directly. Bao et al., 2022

# Masked Autoencoder (MAE)



MAE uses a simpler architecture and no tokenization. He et al. 2021

# Masked Autoencoder - Reconstructions



He et al. 2021

# Masked Autoencoder - Transfer Learning

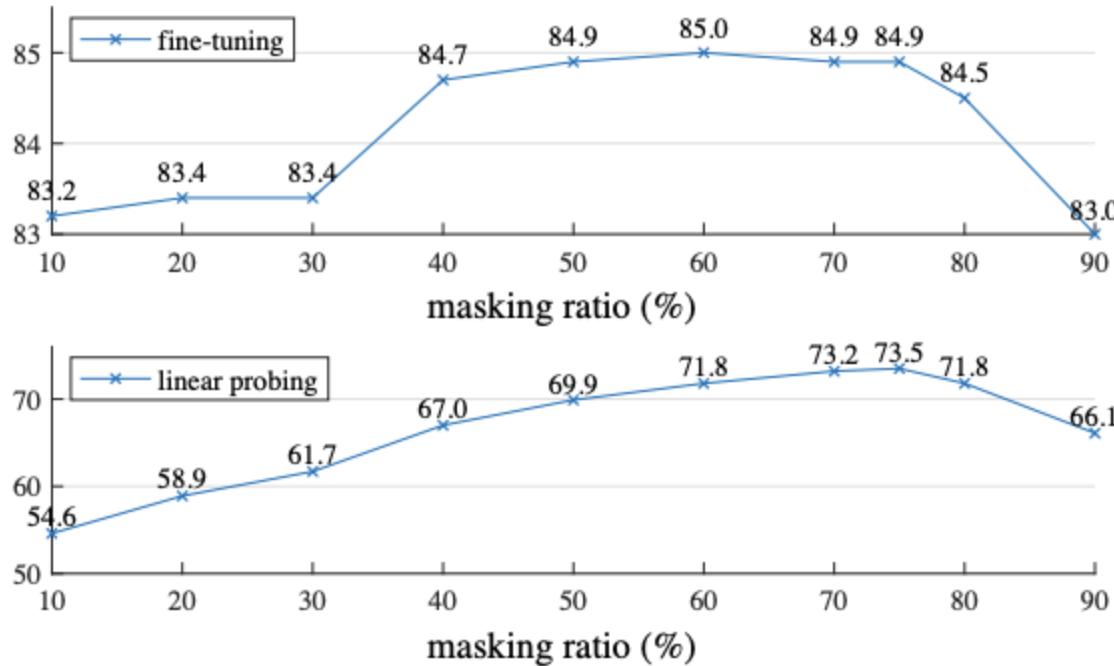


Figure 5. **Masking ratio.** A high masking ratio (75%) works well for both fine-tuning (top) and linear probing (bottom). The y-axes are ImageNet-1K validation accuracy (%) in all plots in this paper.

# Unsupervised Training with Transformers

Transformer architectures make BERT-like masking useful unsupervised pre-training for image-related tasks.

# OTHER APPROACHES

# SimCLR - Contrastive Learning

Basic idea:

- generate two differently augmented versions of the same image
- train a representation that is as similar as possible for the same image, different for different images

Details:

- carefully choose augmentations, watch out for trivial solutions (e.g. color)
- separate representation from scoring
- compute "softmax over cosine similarity" over very large batches
- implemented with ResNet

# SimCLR - Contrastive Learning

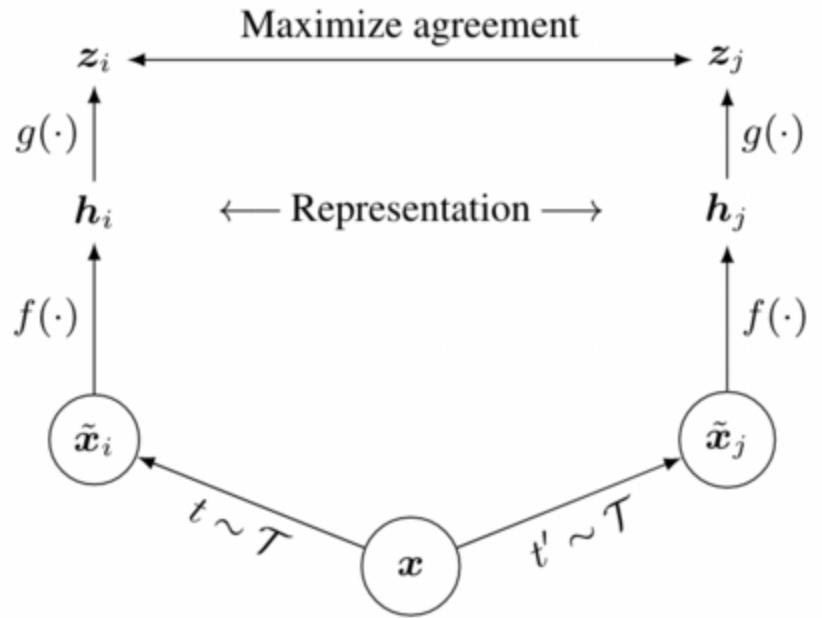


Figure 2. A simple framework for contrastive learning of visual representations. Two separate data augmentation operators are sampled from the same family of augmentations ( $t \sim \mathcal{T}$  and  $t' \sim \mathcal{T}$ ) and applied to each data example to obtain two correlated views. A base encoder network  $f(\cdot)$  and a projection head  $g(\cdot)$  are trained to maximize agreement using a contrastive loss. After training is completed, we throw away the projection head  $g(\cdot)$  and use encoder  $f(\cdot)$  and representation  $\mathbf{h}$  for downstream tasks.

# SimCLR - Augmentations



(a) Original



(b) Crop and resize



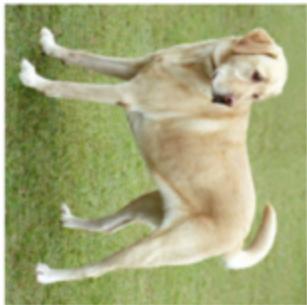
(c) Crop, resize (and flip)



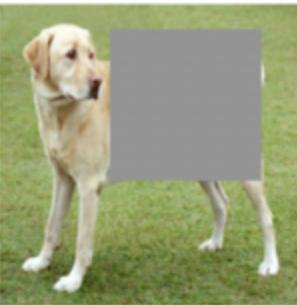
(d) Color distort. (drop)



(e) Color distort. (jitter)



(f) Rotate  $\{90^\circ, 180^\circ, 270^\circ\}$



(g) Cutout



(h) Gaussian noise



(i) Gaussian blur



(j) Sobel filtering

Figure 4. Illustrations of the studied data augmentation operators. Each augmentation can transform data stochastically with some internal parameters (e.g. rotation degree, noise level). Note that we *only* test these operators in ablation, the *augmentation policy used to train our models* only includes *random crop (with flip and resize)*, *color distortion*, and *Gaussian blur*. (Original image cc-by: Von.grzanka)

# SimCLR - Transfer Learning Performance

	Food	CIFAR10	CIFAR100	Birdsnap	SUN397	Cars	Aircraft	VOC2007	DTD	Pets	Caltech-101	Flowers
<i>Linear evaluation:</i>												
SimCLR (ours)	<b>76.9</b>	<b>95.3</b>	80.2	48.4	<b>65.9</b>	60.0	61.2	<b>84.2</b>	<b>78.9</b>	89.2	<b>93.9</b>	<b>95.0</b>
Supervised	75.2	<b>95.7</b>	<b>81.2</b>	<b>56.4</b>	64.9	<b>68.8</b>	<b>63.8</b>	83.8	<b>78.7</b>	<b>92.3</b>	<b>94.1</b>	94.2
<i>Fine-tuned:</i>												
SimCLR (ours)	<b>89.4</b>	<b>98.6</b>	<b>89.0</b>	<b>78.2</b>	<b>68.1</b>	<b>92.1</b>	<b>87.0</b>	<b>86.6</b>	<b>77.8</b>	92.1	<b>94.1</b>	97.6
Supervised	88.7	98.3	<b>88.7</b>	<b>77.8</b>	67.0	91.4	<b>88.0</b>	86.5	<b>78.8</b>	<b>93.2</b>	<b>94.2</b>	<b>98.0</b>
Random init	88.3	96.0	81.9	<b>77.0</b>	53.7	91.3	84.8	69.4	64.1	82.7	72.5	92.5

Table 8. Comparison of transfer learning performance of our self-supervised approach with supervised baselines across 12 natural image classification datasets, for ResNet-50 ( $4\times$ ) models pretrained on ImageNet. Results not significantly worse than the best ( $p > 0.05$ , permutation test) are shown in bold. See Appendix B.8 for experimental details and results with standard ResNet-50.

# DINO

- self-DIstillation with NO labels
- discovers labels / class structure by itself
- unsupervised representation learning for images
- impressive semantic segmentation results
- attention map = segmentation map
- vision transformer or ResNet 50 based

# DINO Architecture

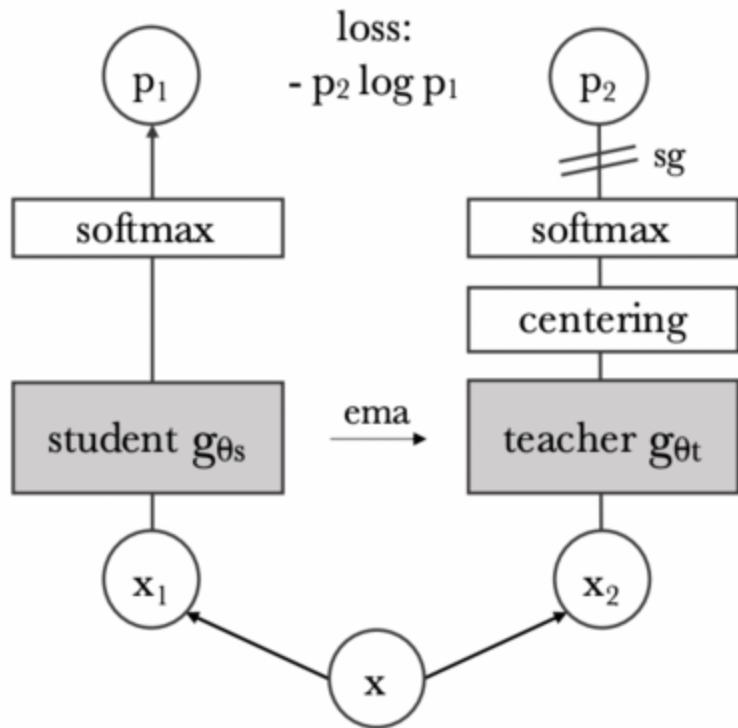


Figure 2: **Self-distillation with no labels.** We illustrate DINO in the case of one single pair of views ( $x_1, x_2$ ) for simplicity. The

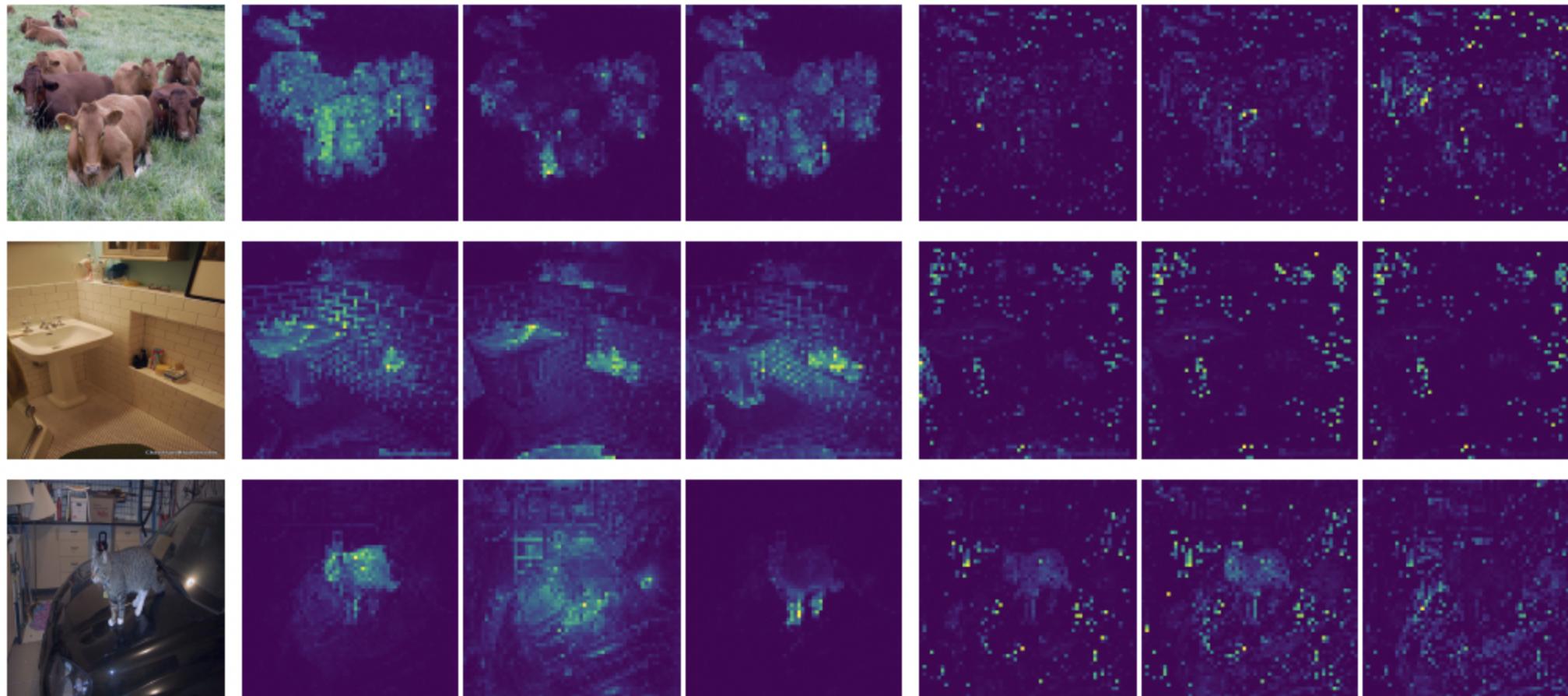
# DINO Results

Method	Arch.	Param.	im/s	Linear	$k$ -NN
Supervised	RN50	23	1237	79.3	79.3
SCLR [12]	RN50	23	1237	69.1	60.7
MoCov2 [15]	RN50	23	1237	71.1	61.9
InfoMin [67]	RN50	23	1237	73.0	65.3
BarlowT [81]	RN50	23	1237	73.2	66.0
OBoW [27]	RN50	23	1237	73.8	61.9
BYOL [30]	RN50	23	1237	74.4	64.8
DCv2 [10]	RN50	23	1237	75.2	67.1
SwAV [10]	RN50	23	1237	<b>75.3</b>	65.7
<b>DINO</b>	RN50	23	1237	<b>75.3</b>	<b>67.5</b>
Supervised	ViT-S	21	1007	79.8	79.8
BYOL* [30]	ViT-S	21	1007	71.4	66.6
MoCov2* [15]	ViT-S	21	1007	72.7	64.4
SwAV* [10]	ViT-S	21	1007	73.5	66.3
<b>DINO</b>	ViT-S	21	1007	<b>77.0</b>	<b>74.5</b>

# DINO Segmentation

DINO

Supervised



# Summary: SimCLR and DINO

High Level:

- SimCLR is a kind of *representation* or *metric* learning
- DINO is a kind of *clustering*
- implementations are complex and with lots of hyperparameters

Conclusions:

- tasks like these may be most useful as additional tasks combined with masking  
(just like BERT)

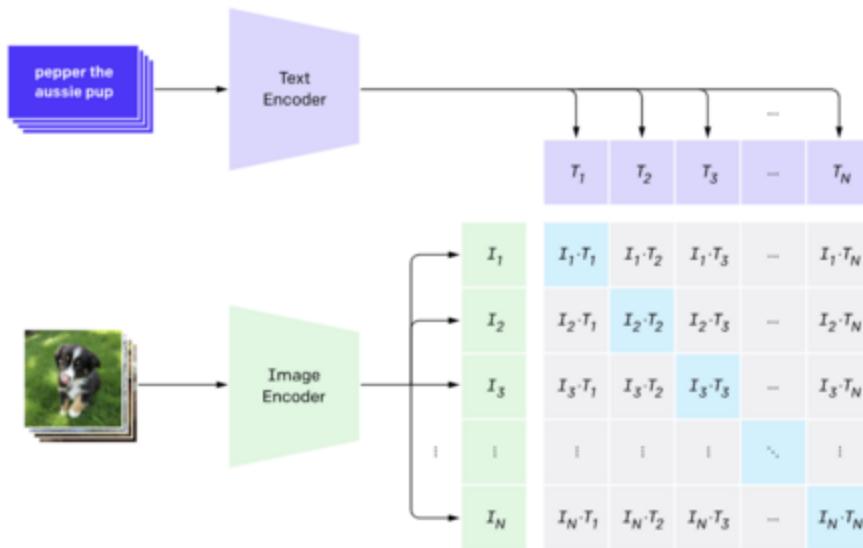
# COMBINING TEXT AND IMAGE MODELS

# Combining Image and Text Models

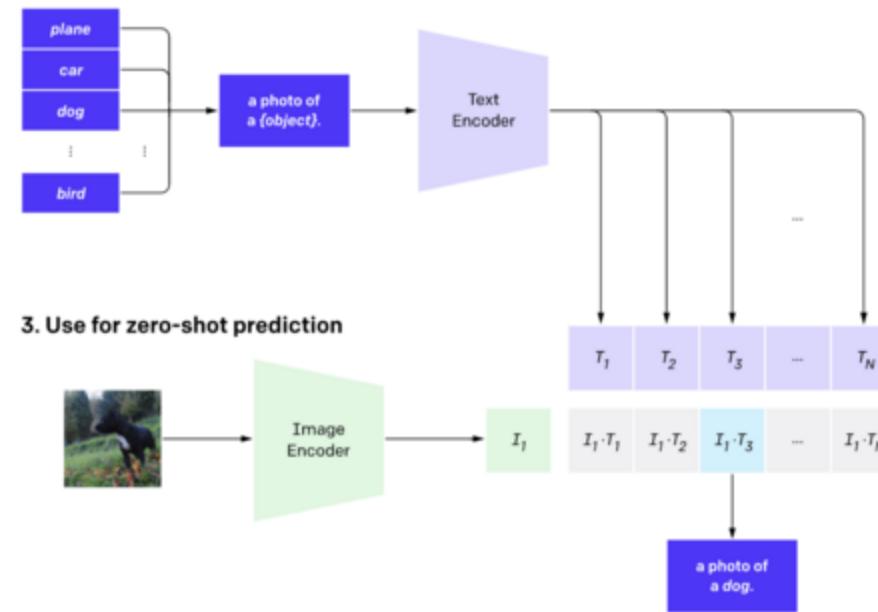
- GPT-3, ExT5, etc. show how natural language models can be used for zero shot learning
- CLIP
  - use natural language supervision for image recognition (weak supervision)
  - vision: transformer or ResNet, language: transformer
  - permit "prompt engineering" to allow different kinds of NLP tasks
  - uses contrastive pretraining (rather than, say, captioning)

# CLIP Architecture

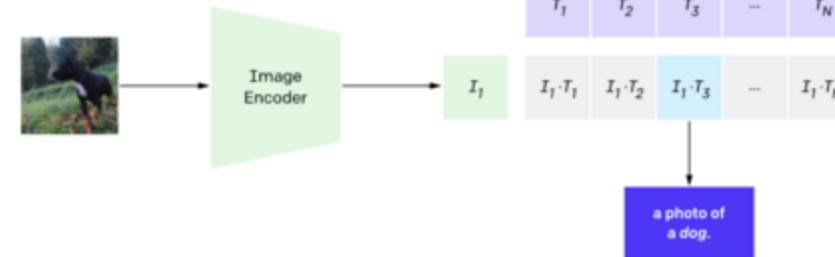
## 1. Contrastive pre-training



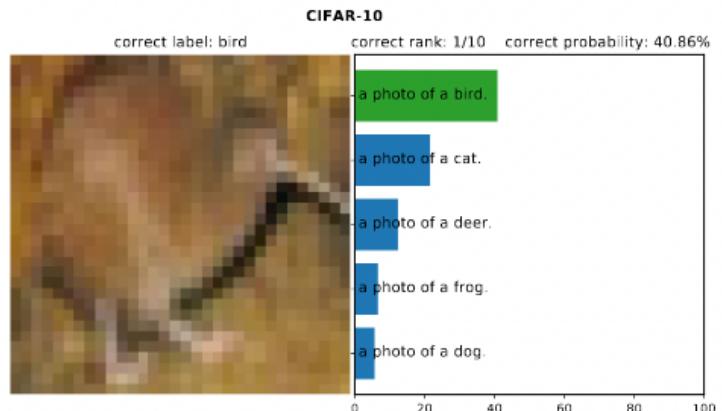
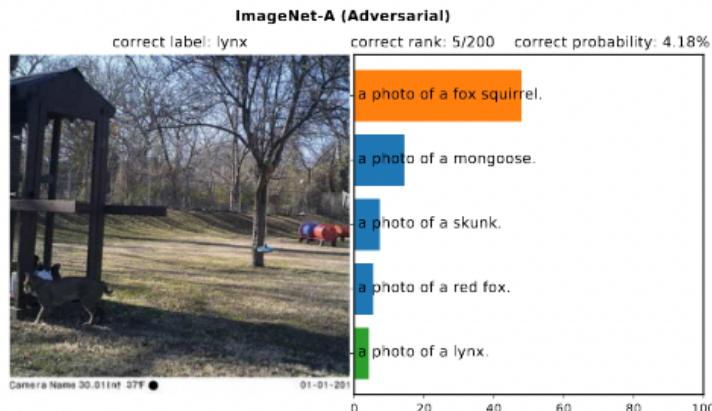
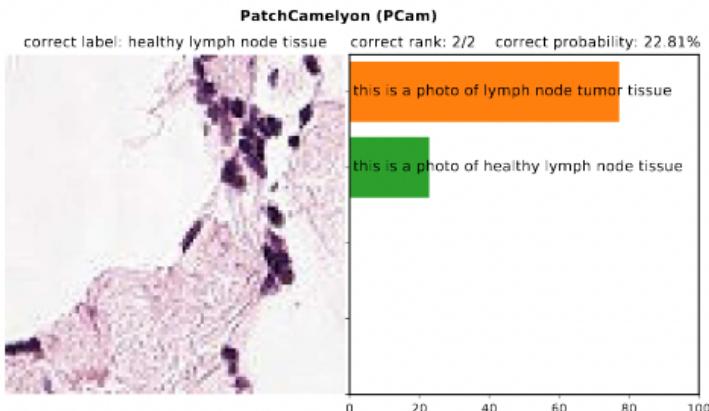
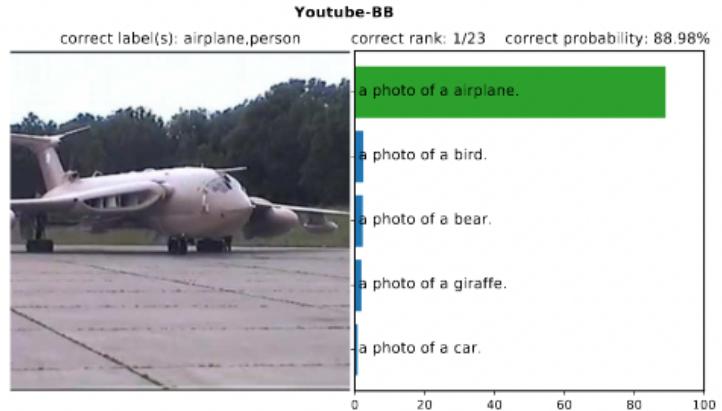
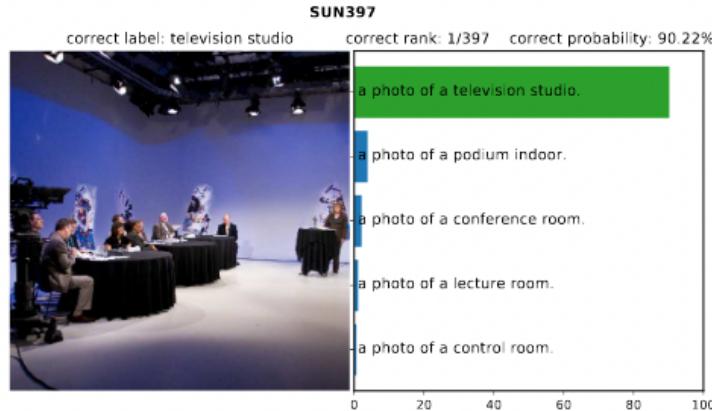
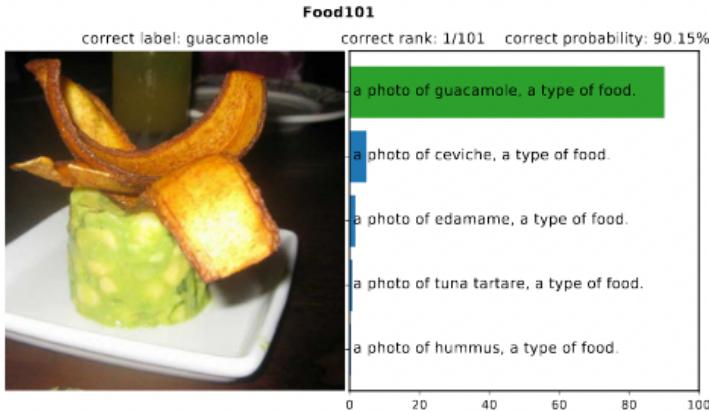
## 2. Create dataset classifier from label text



## 3. Use for zero-shot prediction



# CLIP Results



# CLIP Results

Zero-shot ImageNet accuracy

40%

20%

0%

2M

33M

67M

134M

268M

400M

Images processed

Bag of Words  
Contrastive (CLIP)

Bag of Words  
Prediction

Transformer  
Language Model

4x

3x efficiency

# Discussion

Current and future directions:

- combining text and image
- integrating unsupervised pretrained vision and language models
- integrating video and audio and identifying good self-supervised tasks for these (e.g., VideoMAE, Audio-MAE)
- cross-modal combinations likely also reduce the amount of training data required within each modality

# GENERATIVE AND DENSITY MODELS

# Traditional Approaches

- parametric density estimation
- mixture models
- Bayesian networks
- causal models
- Boltzmann machine
- restricted Boltzmann machine
- Energy Based Models

# Generative and Density Models

Important Applications:

- speech generation, game content generation: conditional generation of content based on some inputs
- image and video enhancement: upscaling, sharpening, interpolation, ray tracing, etc.

Largely Unfulfilled Potential:

- unsupervised learning for transfer learning
- training dataset generation and augmentation
- domain transfer (winter->summer, etc.)

# FLOW MODELS

# Flow Models Idea

- model some complicated  $p(\xi)$
- given  $x \sim p(x) = \mathcal{N}(0, 1)$
- learn an invertible function  $x = f_\theta(\xi)$

# Flow Models - Mapping Densities

Example:

- linear mapping of a standard normal density  $\mathcal{N}(0, 1)$  to a general normal density  $\mathcal{N}(\mu, \Sigma)$
- requires changing the normalization factor b/c linear map is not volume preserving

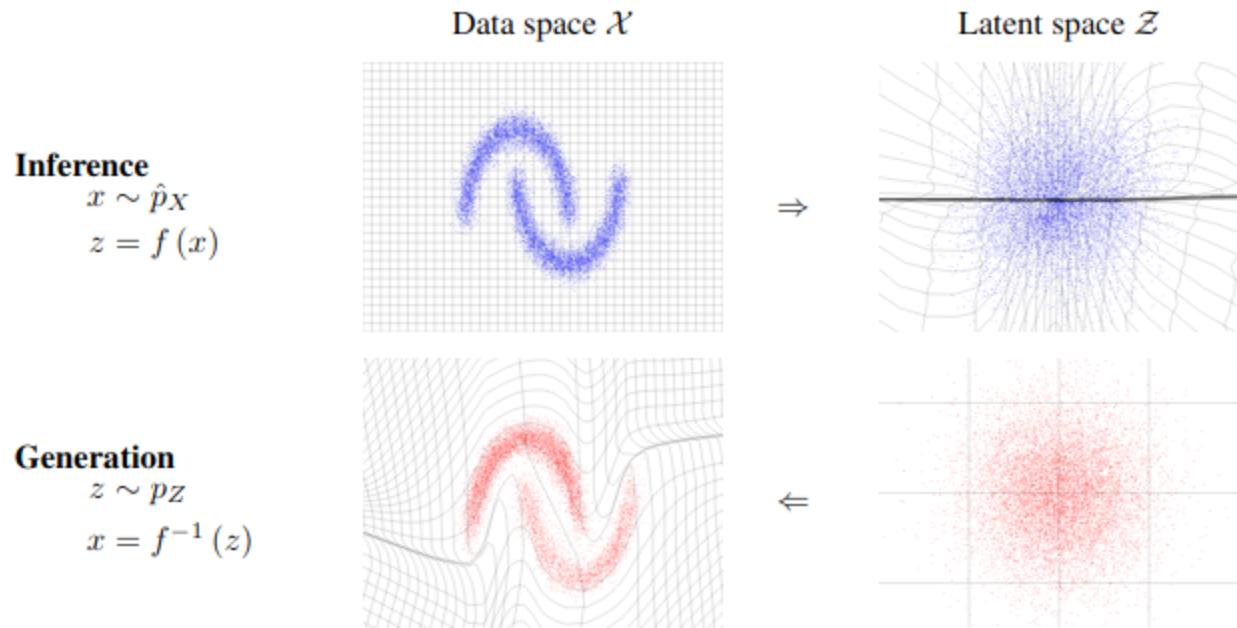
Generalization:

- the local linear relationship between  $x$  and  $\xi$  is given by the Jacobian  $J_{f_\theta}$
- we account for the volume change using the determinant of the Jacobian
- $p(x) = p(\xi) \cdot \det(J_{f_\theta}(\xi))$

Updating the loss function:

- $\arg \min_\theta \mathbb{E}_x[-\log p(x)] = \arg \min_\theta \mathbb{E}_x[-\log p(f_\theta(x)) - \log \det(J_{f_\theta}(\xi))]$

# Flow Models - Illustration

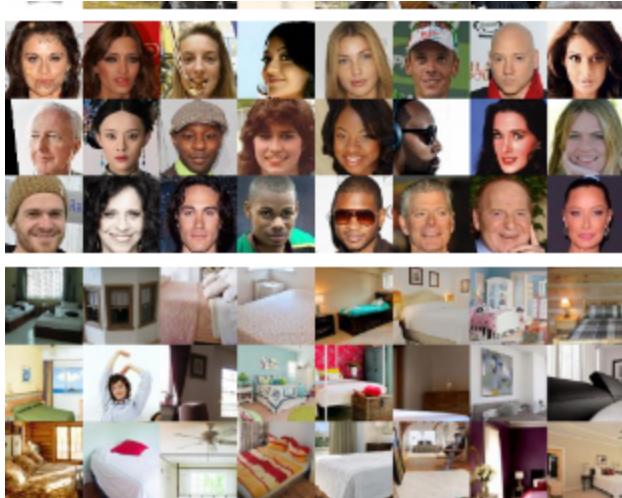


Dinh et al. ICLR 2017

# Flow Models - Technicalities

- mappings compose: if  $f = g \circ h$ 
  - then  $p(x) = p(\xi) \cdot \det(J_g(h(\xi))) \det(J_h(\xi))$
- regular linear layers are too costly to invert, use affine coupling layers instead (NICE/RealNVP)
  - split input to layer in two halves  $x = (x_l, x_h)$
  - $y_l = x_l$
  - $y_h = x_h \odot f_\theta(x_l) + g_\theta(x_l)$  (elementwise affine)
  - the Jacobian is triangular, so its determinant is the product of the diagonal elements
- need to "dequantize" images
- other options: mixtures of logistics, continuous time flows, ...

# Flow Models - Examples



Dinh et al. ICLR 2017

# VAE

# Variational Autoencoder

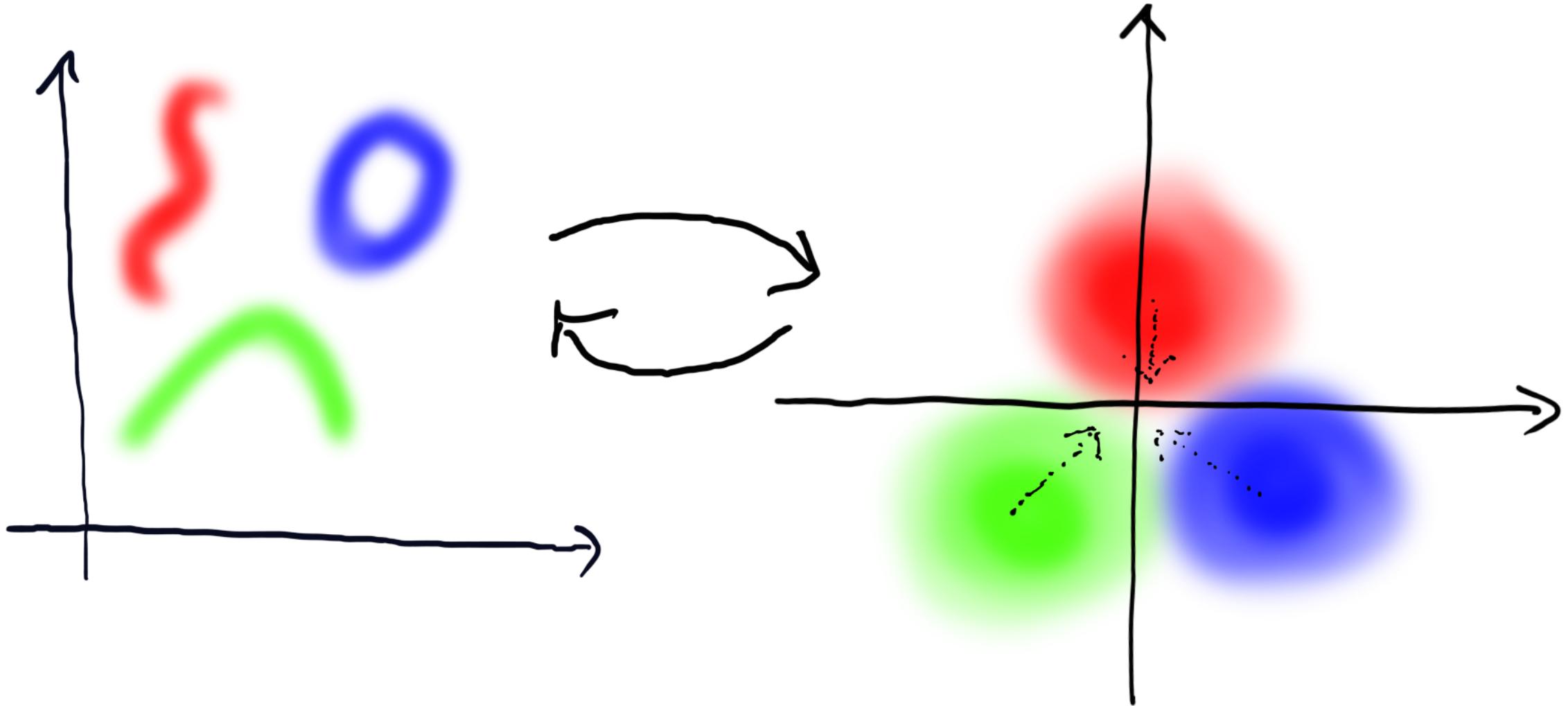
Problem using Autoencoders for Generative Models:

- tends not to generate useful latent space representations
- small changes in latent space not guaranteed to correspond to small changes in output

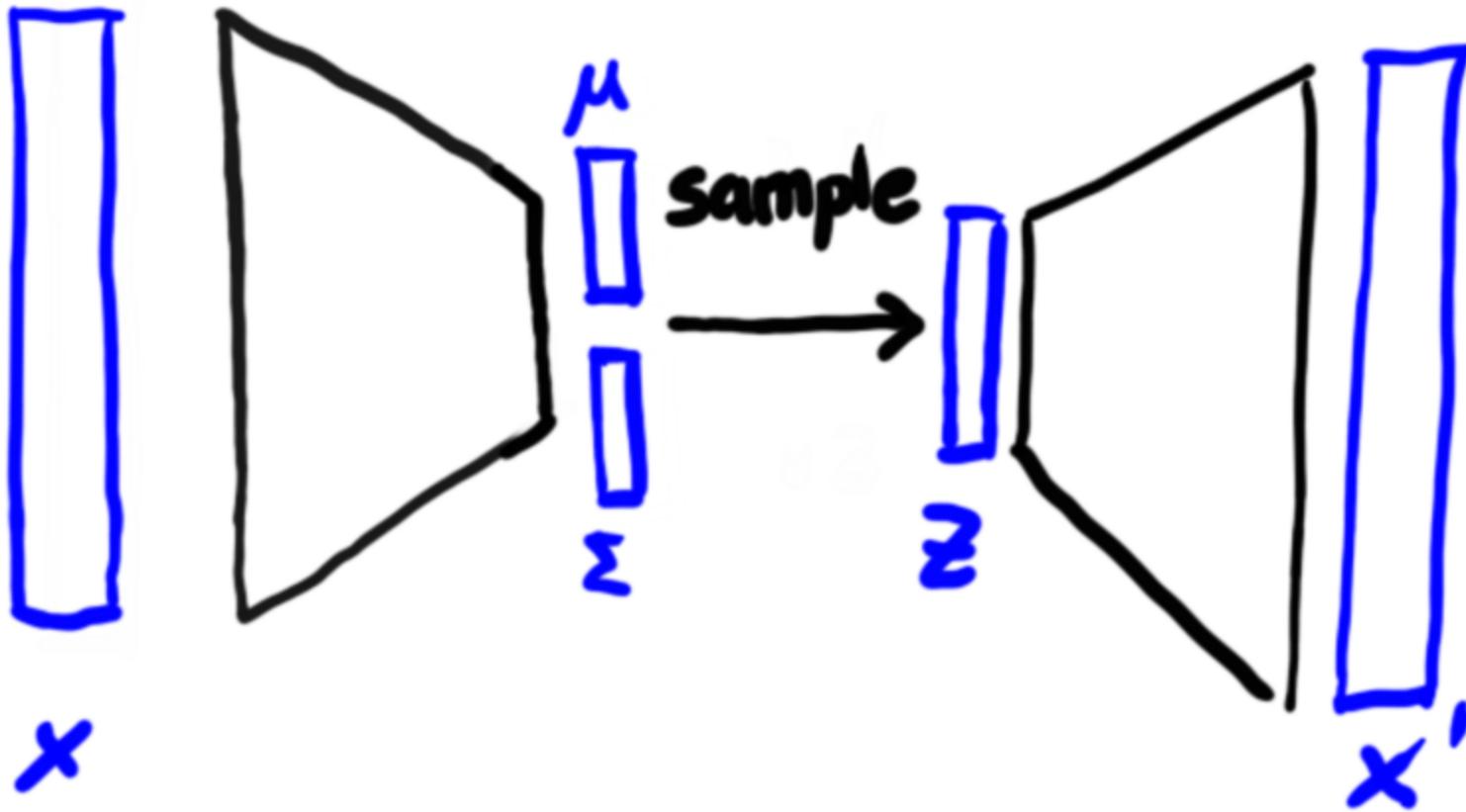
Solution:

- map each input vector  $x$  to a conditional distribution over  $y$
- specifically
  - output parameters for a parametric distribution  $p_\phi(y|x)$
  - choose  $p_\phi(y|x) = \mathcal{N}(y, \mu_x, \Sigma_x)$
  - move  $p_\phi(y|x)$  towards  $\mathcal{N}(0, \mathbf{1})$
- reconstruct  $x$  by sampling from that conditional distribution

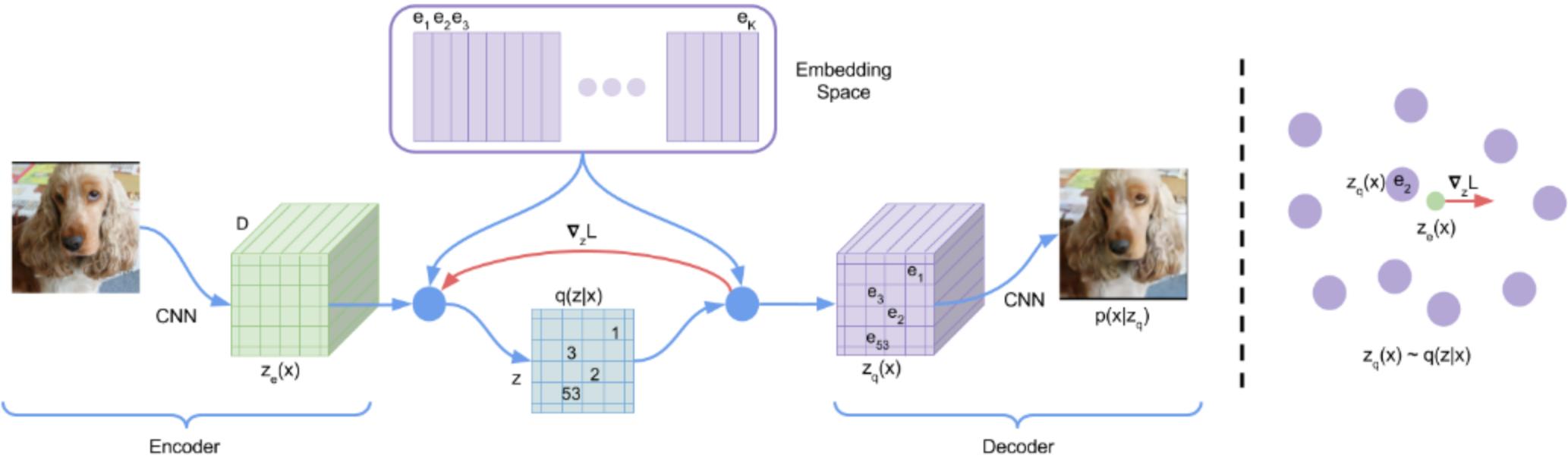
# VAE Objective



# VAE Structure



# VQ-VAE



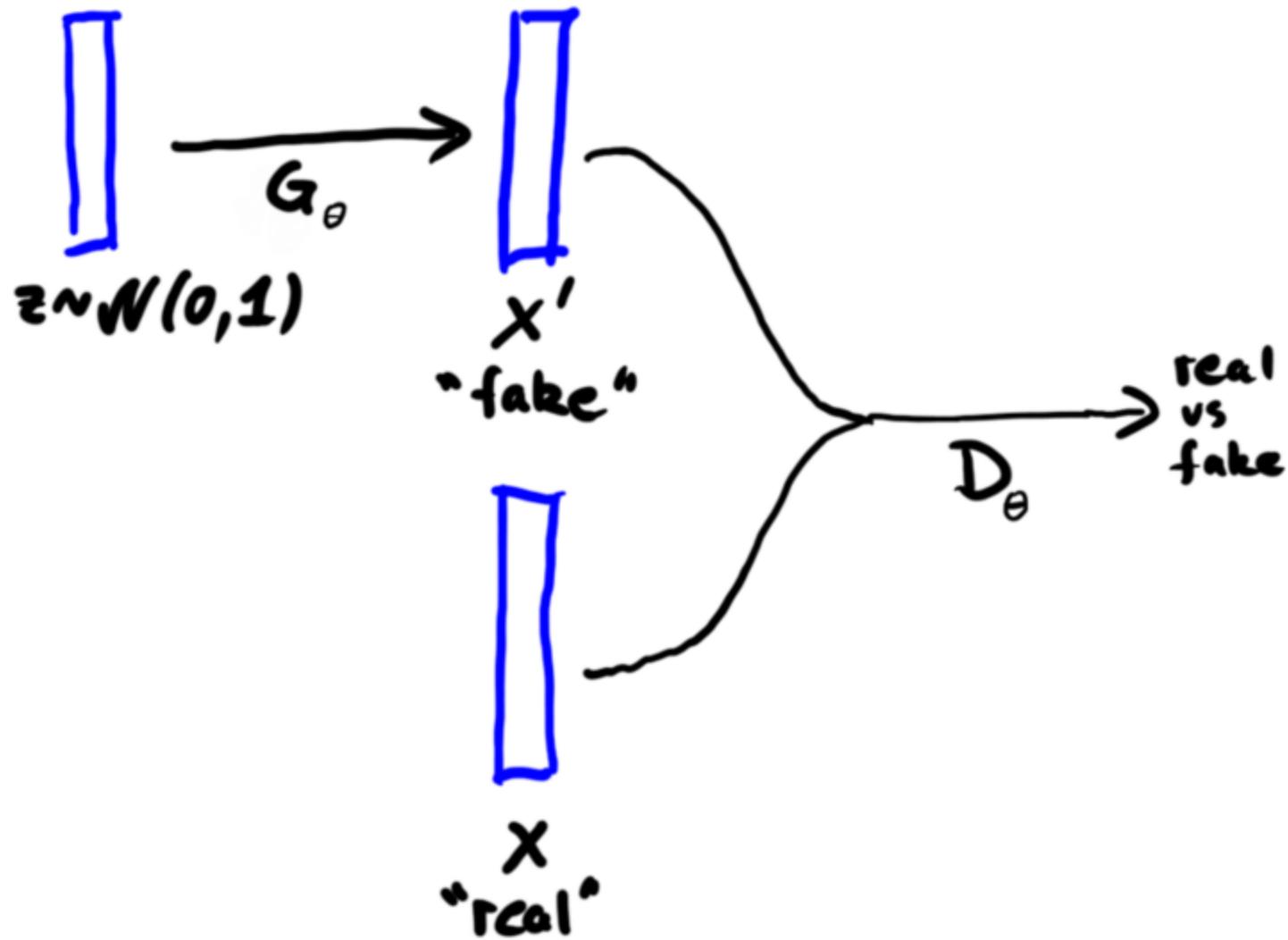
VQ-VAE uses a discrete latent space with a "language model", giving rise to a rather different kind of model; van den Oord et al., Arxiv 2018

# GENERATIVE ADVERSARIAL NETWORKS

# Generative Adversarial Models

- sampling-only models
- sample from an underlying distribution  $z \sim p(z)$
- compute output as  $f_\theta(z)$ , inducing some  $p_\theta(x)$
- for other models, we tried to make  $p_\theta(x)$  close to the data
- with GANs, we cannot access  $p_\theta(x)$ , only sample from it
- a kind of "game" in which a generator tries to fool a discriminator

# GAN



# Major Difficulties Implementing GAN

- low signal from a single global discriminator
  - solution: use patchwise/pixel-wise discrimination
- discriminator easily saturates and generates no usable deltas
  - solution: carefully choose discriminator architecture and limited training
- mode collapse: if  $p(x)$  is multimodal, model may sample a single mode
  - solution: use Wasserstein GAN (roughly: discriminators w/bounded Lipschitz coefficients)

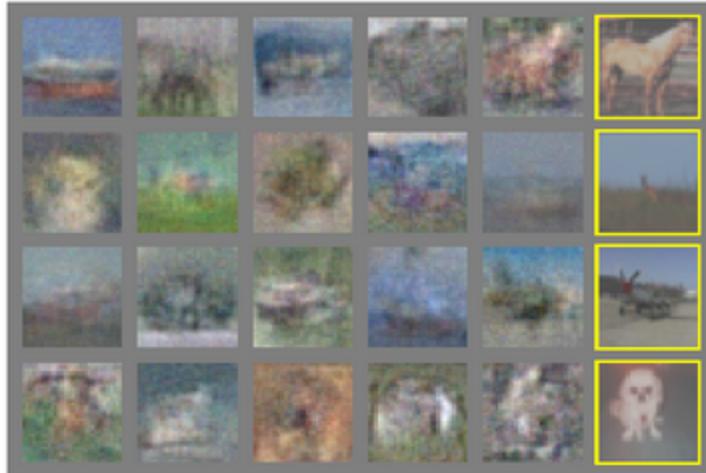
# Original GAN Results



a)



b)

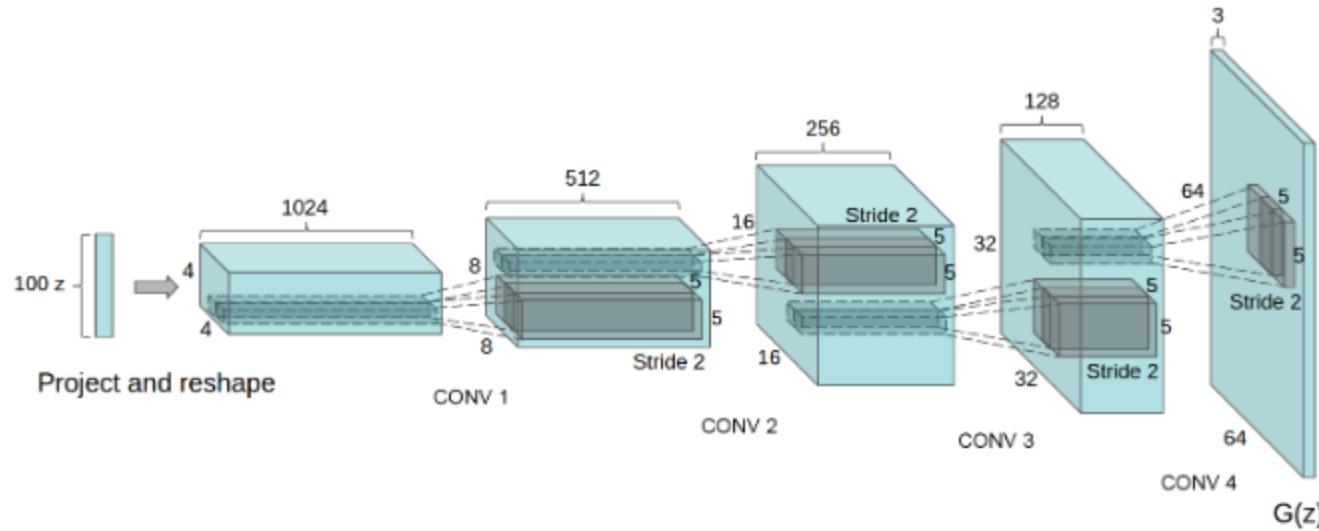


c)



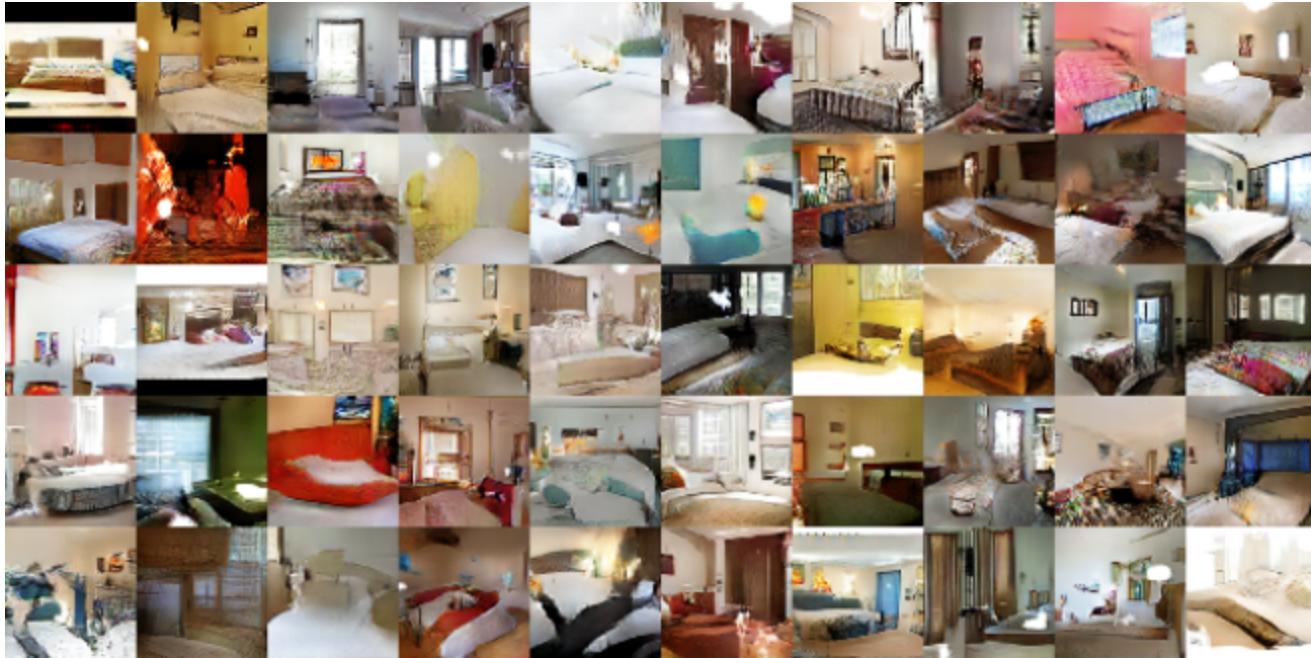
d)

# Deep Convolutional GAN (DCGAN)



Radford et al., 2016; Arxiv 1511.06434

# DCGAN - Results



Radford et al., 2016; Arxiv 1511.06434

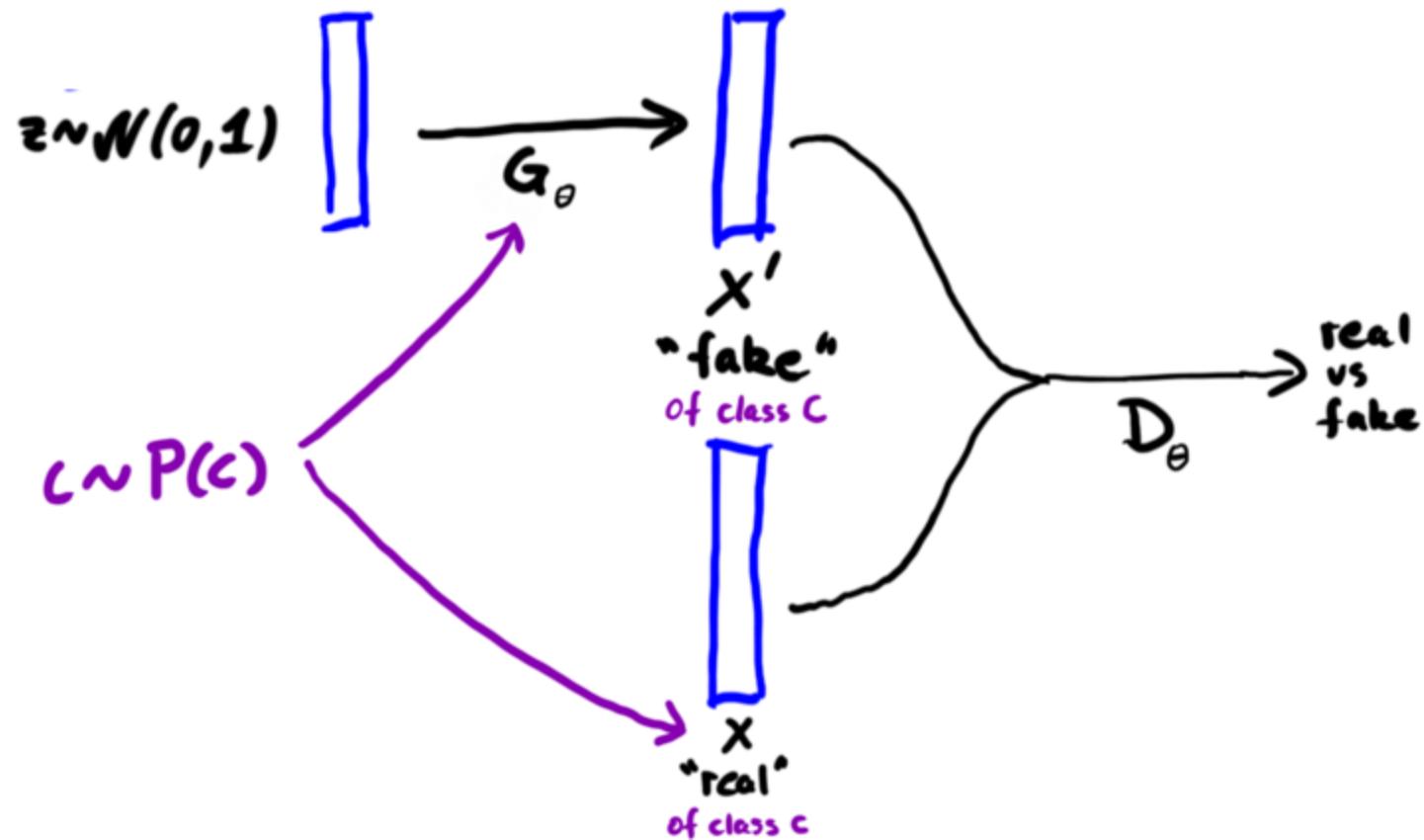
# DCGAN - Representation Learning / Features

Table 1: CIFAR-10 classification results using our pre-trained model. Our DCGAN is not pre-trained on CIFAR-10, but on Imagenet-1k, and the features are used to classify CIFAR-10 images.

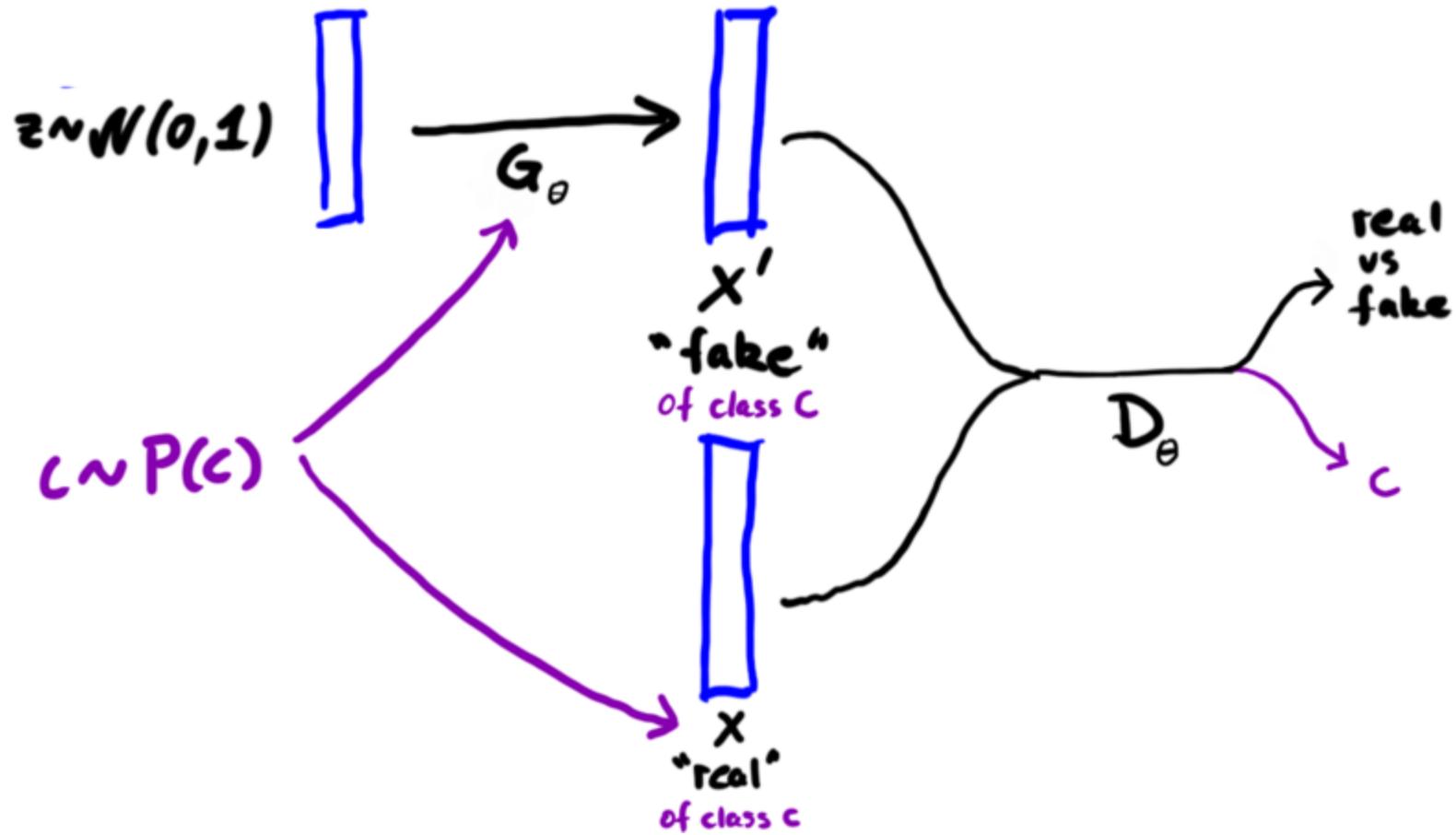
Model	Accuracy	Accuracy (400 per class)	max # of features units
1 Layer K-means	80.6%	63.7% ( $\pm 0.7\%$ )	4800
3 Layer K-means Learned RF	82.0%	70.7% ( $\pm 0.7\%$ )	3200
View Invariant K-means	81.9%	72.6% ( $\pm 0.7\%$ )	6400
Exemplar CNN	84.3%	77.4% ( $\pm 0.2\%$ )	1024
DCGAN (ours) + L2-SVM	82.8%	73.8% ( $\pm 0.4\%$ )	512

Radford et al., 2016; Arxiv 1511.06434

# Conditional GAN

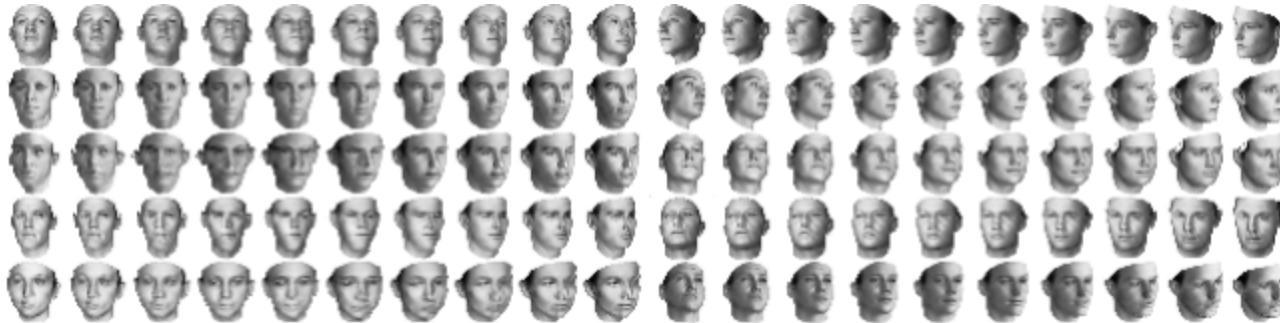


# InfoGAN



Chen et al., 2016; arXiv:1606.03657.

# InfoGAN - Disentangling Latent Codes



(a) Azimuth (pose)

(b) Elevation



(c) Lighting

(d) Wide or Narrow

Chen et al., 2016; arXiv:1606.03657.

# BigGAN Results



Large bag of tricks. Brock et al. 2018.

# DISTRIBUTION ALIGNMENT

# Problem

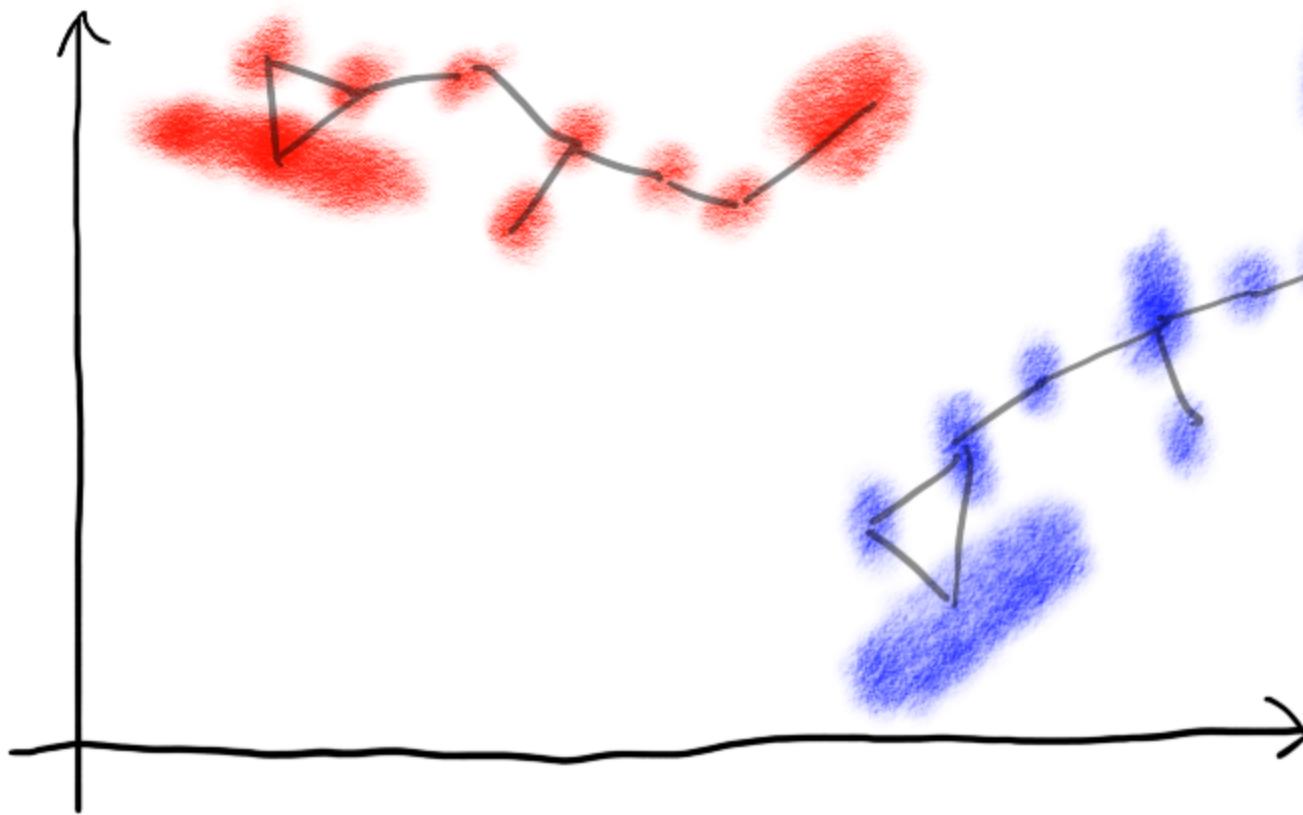
Assume we have two different but related distributions  $p(x)$  and  $p'(x)$  but no corresponding samples.

Examples:

- daytime vs nighttime images
- outdoors scenes in sunshine vs rain
- photographic images vs oil paintings
- clean binary document image vs photographic capture

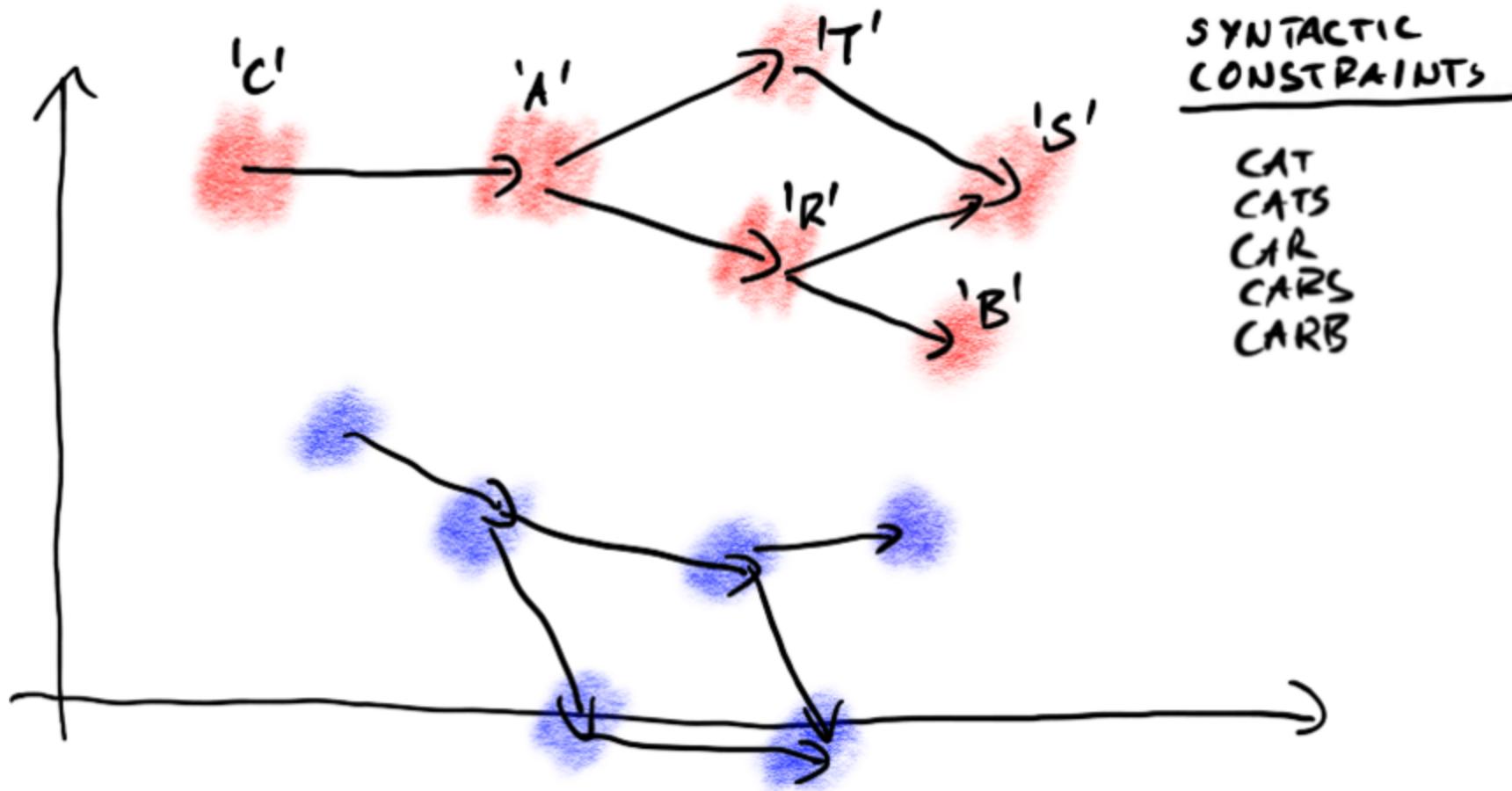
Can we map between these two domains given just independent samples from each?  
Sometimes.

# Alignment of Mixture Densities (no other info)



- distribution alignment is possible for distributions with a lot of structure
- ... if the "general shape" of the distribution is preserved

# Alignment of Mixture Densities (with sequences)



- syntactic constraints (e.g. sequences of samples) help with distribution alignment
- this is what EM training for OCR, speech, etc. tends to do

# Uses of Distribution Alignment for Training

E.g. AV images: summer and winter, lots of summer training data, little winter training data

Approach 1:

- train model on summer + translate winter to summer

Approach 2:

- use distribution alignment to generate lots of winter images from summer images, then augment the training set with that
- train model on summer images and artificially generated winter images

# CycleGAN

- two generators  $G' : x \rightarrow x'$ ,  $G : x' \rightarrow x$
- two discriminators:  $D'$  (discriminates real/fake  $x'$ ),  $D$  (discriminates real/fake  $x$ )
- train  $G, D$  and  $G', D'$  like a regular GAN
- require that  $G' \circ G$  and  $G \circ G'$  behave like identities

# CycleGAN Examples

Monet  $\leftrightarrow$  Photos



Monet  $\rightarrow$  photo

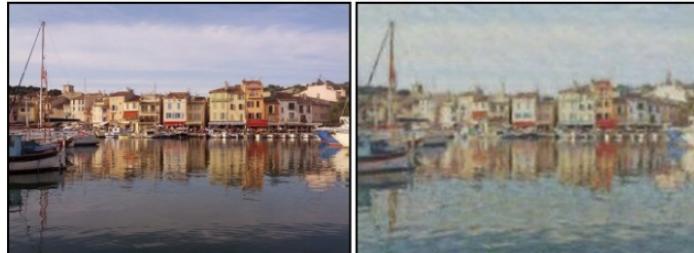


photo  $\rightarrow$  Monet

Zebras  $\leftrightarrow$  Horses



zebra  $\rightarrow$  horse



horse  $\rightarrow$  zebra

Summer  $\leftrightarrow$  Winter



summer  $\rightarrow$  winter



winter  $\rightarrow$  summer

# CycleGAN Examples

EMOIR OF THE AUTHOR.

his thought, that nothing is i  
; of Christ to engage in, i  
effectually promote the ki  
laker. Perhaps it is not p  
d the world will hardly belie  
een taken in composing th  
what care I have endeav

 $d_A$ 

$$g_{AB} \longrightarrow$$

$$\longleftarrow g_{BA}$$

EMOIR OF THE AUTHOR.

his thought, that nothing is i  
; of Christ to engage in, i  
effectually promote the ki  
laker. Perhaps it is not p  
d the world will hardly belie  
een taken in composing th  
what care I have endeav

 $d_B$

# Ongoing and Future Research

- extensions to text, speech, music, video, etc.
- condition on language (of course, already lots of news coverage)
- engineering work to remove artifacts and make output indistinguishable from real
- create forensic tools that can detect fakes
- combine transformers with each of the different generative approaches
- use non-generative unsupervised pretraining to help improve generative models

# Limitations

It is an open question whether generative models will ever be useful unsupervised models for transfer learning to discriminative or other generative models.

High quality generative output requires modeling a large amount of detail that is irrelevant to most tasks.

- Almahairi, Amjad, Sai Rajeshwar, Alessandro Sordoni, Philip Bachman, and Aaron Courville. "Augmented CycleGAN: Learning Many-to-Many Mappings from Unpaired Data." In *Proceedings of the 35th International Conference on Machine Learning*, 195–204. PMLR, 2018. <https://proceedings.mlr.press/v80/almahairi18a.html>.
- Arjovsky, Martin, Soumith Chintala, and Léon Bottou. "Wasserstein GAN." arXiv, December 6, 2017. <https://doi.org/10.48550/arXiv.1701.07875>.
- Bao, Hangbo, Li Dong, Songhao Piao, and Furu Wei. "BEiT: BERT Pre-Training of Image Transformers." arXiv, September 3, 2022. <https://doi.org/10.48550/arXiv.2106.08254>.
- Bell, Anthony J., and Terrence J. Sejnowski. "An Information-Maximization Approach to Blind Separation and Blind Deconvolution." *Neural Computation* 7 (1995): 1129–59.
- BELL, ANTHONY J., and TERRENCE J. SEJNOWSKI. "The 'Independent Components' of Natural Scenes Are Edge Filters." *Vision Research* 37, no. 23 (December 1997): 3327–38.
- Brock, Andrew, Jeff Donahue, and Karen Simonyan. "Large Scale GAN Training for High Fidelity Natural Image Synthesis." arXiv, February 25, 2019. <https://doi.org/10.48550/arXiv.1809.11096>.
- Bromley, Jane, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. "Signature Verification Using a 'Siamese' Time Delay Neural Network," n.d., 8.
- Brown, Tom B., Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, et al. "Language Models Are Few-Shot Learners." arXiv, July 22, 2020. <https://doi.org/10.48550/arXiv.2005.14165>.
- . "Language Models Are Few-Shot Learners." arXiv, July 22, 2020. <https://doi.org/10.48550/arXiv.2005.14165>.
- Caron, Mathilde, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. "Emerging Properties in Self-Supervised Vision Transformers." arXiv, May 24, 2021. <https://doi.org/10.48550/arXiv.2104.14294>.
- Chen, Ting, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. "A Simple Framework for Contrastive Learning of Visual Representations." arXiv, June 30, 2020. <https://doi.org/10.48550/arXiv.2002.05709>.
- Chen, Ting, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey Hinton. "Big Self-Supervised Models Are Strong Semi-Supervised Learners." arXiv, October 25, 2020. <https://doi.org/10.48550/arXiv.2006.10029>.
- Chen, Xi, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. "InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets." arXiv, June 11, 2016. <https://doi.org/10.48550/arXiv.1606.03657>.
- Creswell, Antonia, Tom White, Vincent Dumoulin, Kai Arulkumaran, Biswa Sengupta, and Anil A. Bharath. "Generative Adversarial Networks: An Overview." *IEEE Signal Processing Magazine* 35, no. 1 (January 2018): 53–65. <https://doi.org/10.1109/MSP.2017.2765202>.
- Croitoru, Florinel-Alin, Vlad Hondru, Radu Tudor Ionescu, and Mubarak Shah. "Diffusion Models in Vision: A Survey." arXiv, September 10, 2022. <https://doi.org/10.48550/arXiv.2209.04747>.
- Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. "BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding." arXiv, May 24, 2019. <https://doi.org/10.48550/arXiv.1810.04805>.
- Dinh, Laurent, David Krueger, and Yoshua Bengio. "NICE: Non-Linear Independent Components Estimation." arXiv, April 10, 2015. <https://doi.org/10.48550/arXiv.1410.8516>.
- Dinh, Laurent, Jascha Sohl-Dickstein, and Samy Bengio. "Density Estimation Using Real NVP." arXiv, February 27, 2017. <https://doi.org/10.48550/arXiv.1605.08803>.
- Doersch, Carl, Abhinav Gupta, and Alexei A. Efros. "Unsupervised Visual Representation Learning by Context Prediction." arXiv, January 16, 2016. <https://doi.org/10.48550/arXiv.1505.05192>.
- Dosovitskiy, Alexey, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, et al. "An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale." arXiv, June 3, 2021. <https://doi.org/10.48550/arXiv.2010.11929>.
- Du, Yilun, and Igor Mordatch. "Implicit Generation and Generalization in Energy-Based Models." arXiv, June 29, 2020. <https://doi.org/10.48550/arXiv.1903.08689>.
- Goldberg, Yoav, and Omer Levy. "Word2vec Explained: Deriving Mikolov et al.'s Negative-Sampling Word-Embedding Method." arXiv, February 15, 2014. <https://doi.org/10.48550/arXiv.1402.3722>.
- Goodfellow, Ian J., Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. "Generative Adversarial Networks." arXiv, June 10, 2014. <https://doi.org/10.48550/arXiv.1406.2661>.
- Graves, Alex. "Generating Sequences With Recurrent Neural Networks." arXiv, June 5, 2014. <http://arxiv.org/abs/1308.0850>.
- Grill, Jean-Bastien, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, et al. "Bootstrap Your Own Latent: A New Approach to Self-Supervised Learning." arXiv, September 10, 2020. <https://doi.org/10.48550/arXiv.2006.07733>.
- He, Kaiming, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. "Masked Autoencoders Are Scalable Vision Learners." arXiv, December 19, 2021. <https://doi.org/10.48550/arXiv.2111.06377>.
- OpenAI. "Implicit Generation and Generalization Methods for Energy-Based Models," March 21, 2019. <https://openai.com/blog/energy-based-models/>.
- Jang, Eric, Shixiang Gu, and Ben Poole. "Categorical Reparameterization with Gumbel-Softmax." arXiv, August 5, 2017. <https://doi.org/10.48550/arXiv.1611.01144>.
- Kingma, Diederik P., and Max Welling. "An Introduction to Variational Autoencoders." *Foundations and Trends® in Machine Learning* 12, no. 4 (2019): 307–92. <https://doi.org/10.1561/2200000056>.
- . "An Introduction to Variational Autoencoders." *Foundations and Trends® in Machine Learning* 12, no. 4 (2019): 307–92. <https://doi.org/10.1561/2200000056>.
- . "Auto-Encoding Variational Bayes." arXiv, May 1, 2014. <https://doi.org/10.48550/arXiv.1312.6114>.
- Koch, Gregory, Richard Zemel, and Ruslan Salakhutdinov. "Siamese Neural Networks for One-Shot Image Recognition,"

- LeCun, Yann. "Learning Invariant Feature Hierarchies." In *Computer Vision - ECCV 2012. Workshops and Demonstrations*, edited by Andrea Fusiello, Vittorio Murino, and Rita Cucchiara, 7583:496–505. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012. [https://doi.org/10.1007/978-3-642-33863-2\\_51](https://doi.org/10.1007/978-3-642-33863-2_51).
- Makansi, Osama, Eddy Ilg, Özgün Çicek, and Thomas Brox. "Overcoming Limitations of Mixture Density Networks: A Sampling and Fitting Framework for Multimodal Future Prediction." In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 7137–46, 2019. <https://doi.org/10.1109/CVPR.2019.00731>.
- McCann, Bryan, James Bradbury, Caiming Xiong, and Richard Socher. "Learned in Translation: Contextualized Word Vectors." arXiv, June 20, 2018. <https://doi.org/10.48550/arXiv.1708.00107>.
- Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean. "Efficient Estimation of Word Representations in Vector Space." arXiv, September 6, 2013. <https://doi.org/10.48550/arXiv.1301.3781>.
- Oord, Aaron van den, Oriol Vinyals, and Koray Kavukcuoglu. "Neural Discrete Representation Learning." arXiv, May 30, 2018. <http://arxiv.org/abs/1711.00937>.
- Papamakarios, George, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. "Normalizing Flows for Probabilistic Modeling and Inference." arXiv, April 8, 2021. <https://doi.org/10.48550/arXiv.1912.02762>.
- Park, Taesung, Alexei A. Efros, Richard Zhang, and Jun-Yan Zhu. "Contrastive Learning for Unpaired Image-to-Image Translation." arXiv, August 20, 2020. <https://doi.org/10.48550/arXiv.2007.15651>.
- Pathak, Deepak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A. Efros. "Context Encoders: Feature Learning by Inpainting." arXiv, November 21, 2016. <https://doi.org/10.48550/arXiv.1604.07379>.
- Pennington, Jeffrey, Richard Socher, and Christopher Manning. "Glove: Global Vectors for Word Representation." In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1532–43. Doha, Qatar: Association for Computational Linguistics, 2014. <https://doi.org/10.3115/v1/D14-1162>.
- Pentland, Alex. "E g e d c e s for Recognition," n.d., 16.
- Peters, Matthew E., Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. "Deep Contextualized Word Representations." arXiv, March 22, 2018. <http://arxiv.org/abs/1802.05365>.
- Principe, Jose C., Dongxin Xu, and John W. Fisher III. "Information-Theoretic Learning," 1999.
- Radford, Alec, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. "Improving Language Understanding by Generative Pre-Training," n.d., 12.
- Razavi, Ali, Aaron van den Oord, and Oriol Vinyals. "Generating Diverse High-Fidelity Images with VQ-VAE-2." arXiv, June 2, 2019. <https://doi.org/10.48550/arXiv.1906.00446>.
- Rocca, Joseph. "Understanding Variational Autoencoders (VAEs)." Medium, March 21, 2021. <https://towardsdatascience.com/understanding-variational-autoencoders-vae-f70510919f73>.
- Shree, Priya. "The Journey of Open AI GPT Models." Walmart Global Tech Blog (blog), November 10, 2020. <https://medium.com/walmartglobaltech/the-journey-of-open-ai-gpt-models-32d95b7b7fb2>.
- Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. "Attention Is All You Need." arXiv, December 5, 2017. <https://doi.org/10.48550/arXiv.1706.03762>.
- Vincent, Pascal, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. "Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion," n.d., 38.
- Zhang, Han, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. "Self-Attention Generative Adversarial Networks." arXiv, June 14, 2019. <https://doi.org/10.48550/arXiv.1805.08318>.
- Zhang, Richard, Phillip Isola, and Alexei A. Efros. "Split-Brain Autoencoders: Unsupervised Learning by Cross-Channel Prediction." arXiv, April 20, 2017. <https://doi.org/10.48550/arXiv.1611.09842>.
- Zhu, Jun-Yan, Taesung Park, Phillip Isola, and Alexei A. Efros. "Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks." arXiv, August 24, 2020. <https://doi.org/10.48550/arXiv.1703.10593>.