Perceptual Loss, GANs (part I)

Jun-Yan Zhu

16-726 Learning-based Image Synthesis, Spring 2022

many slides from Alyosha Efros, Phillip Isola, Richard Zhang, James Hays, and Andrea Vedaldi, Jitendra Malik.
HW1 (hints)
Template matching

- **Goal:** find 🎟️ in image
- **Main challenge:** What is a good similarity or distance measure between two patches?
  - Correlation
  - Zero-mean correlation
  - Sum Square Difference
  - Normalized Cross Correlation
Matching with filters

- Goal: find \( \text{eye} \) in image
- Method 0: filter the image with eye patch

\[
 h[m,n] = \sum_{k,l} g[k,l] f[m+k, n+l]
\]

What went wrong?

Input

Filtered Image
Matching with filters

- Goal: find 👀 in image
- Method 1: filter the image with zero-mean eye

\[ h[m, n] = \sum_{k, l} (f[k, l] - \bar{f}) \left( g[m + k, n + l] \right) \]

\( f = \text{image} \)
\( g = \text{filter} \)
Matching with filters

• Goal: find an eye in image

• Method 2: SSD (Sum Square Difference)

\[ h[m, n] = \sum_{k,l} (g[k, l] - f[m + k, n + l])^2 \]

\( f = \text{image} \)
\( g = \text{filter} \)

Input
1- sqrt(SSD)
Thresholded Image

True detections
Matching with filters

\[ h[m,n] = \sum_{k,l} (g[k,l] - f[m+k,n+l])^2 \]

- Can SSD be implemented with linear filters?

f = image

g = filter
Matching with filters

- Goal: find \(\text{\(\bigcirc\)}\) in image

- Method 2: SSD (Sum Square Difference)

\[
h[m,n] = \sum_{k,l} (g[k,l] - f[m+k,n+l])^2
\]

\(f = \text{image}\)
\(g = \text{filter}\)

What’s the potential downside of SSD?
Matching with filters

- Goal: find in image
- Method 2: Normalized Cross-Correlation

\[
h[m,n] = \frac{\sum_{k,l} (g[k,l] - \bar{g})(f[m+k,n+l] - \bar{f}_{m,n})}{\left(\sum_{k,l} (g[k,l] - \bar{g})^2 \sum_{k,l} (f[m+k,n+l] - \bar{f}_{m,n})^2\right)^{0.5}}
\]
Matching with filters

- **Goal:** find $\in$ in image
- **Method 2:** Normalized Cross-Correlation
Matching with filters

- Goal: find in image
- Method 2: Normalized Cross-Correlation
Q: What is the best method to use?

• Answer: Depends

• Zero-mean filter: fastest but not a great matcher

• SSD: next fastest, sensitive to overall intensity

• Normalized cross-correlation: slowest, invariant to local average intensity and contrast
Review
(CNN for Image Synthesis)
Computer Vision before 2012

Features ➔ Clustering ➔ Pooling ➔ Classification ➔ Cat
Computer Vision Now


[Image: Diagram of computer vision process involving features, clustering, pooling, classification, and deep net for identifying a cat]
Deep Learning for Computer Vision

Top 5 accuracy on ImageNet benchmark

- [Redmon et al., 2018] Object detection
- [Güler et al., 2018] Human understanding
- [Zhao et al., 2017] Autonomous driving
Can Deep Learning Help Graphics?

Cat \(\rightarrow\) Modeling \(\rightarrow\) Texturing \(\rightarrow\) Lighting \(\rightarrow\) Rendering

Modeling

Texturing

Lighting

Rendering
Can Deep Learning Help Graphics?

Cat ➔ Modeling ➔ Texturing ➔ Lighting ➔ Rendering

Did not work
Generating images is hard!

Cat ➔ Modeling ➔ Texturing ➔ Lighting ➔ Rendering

8 ➔ Deep Net ➔ 28x28 pixels
Better Architectures
Fractionally-strided Convolution

Regular conv  Fractiaionally-strided conv
Generating chairs conditional on chair ID, viewpoint, and transformation parameters

Dosovitskiy et al. Learning to Generate Chairs, Tables and Cars with Convolutional Networks
PAMI 2017 (CVPR 2015)
With Varying Viewpoints

Dosovitskiy et al. Learning to Generate Chairs, Tables and Cars with Convolutional Networks
PAMI 2017 (CVPR 2015)
With Varying Transformation Parameters

Dosovitskiy et al. Learning to Generate Chairs, Tables and Cars with Convolutional Networks
PAMI 2017 (CVPR 2015)
Interpolation between Two Chairs

Dosovitskiy et al. Learning to Generate Chairs, Tables and Cars with Convolutional Networks
PAMI 2017 (CVPR 2015)
Better Loss Functions
Simple L2 regression doesn’t work 😞
What is a good objective \( \mathcal{L} \)?
- Capture realism
- Calculate image distance
- Adapt to new tasks/data.

\[
\arg \min_G \mathcal{L}(G(x), y)
\]

**Problem Statement**

Loss function

**Input Image**
**Generator**
**Output Image**

**Learnable rendering**
Designing Loss Functions

L2 regression

\[
\text{arg min}_G \mathbb{E}_{(x,y)}[||G(x) - y||]
\]
Designing Loss Functions

Image colorization

Super-resolution

L2 regression

L2 regression

Slide credit: Phillip Isola
Designing Loss Functions

Image colorization

Classification Loss:
Cross entropy objective, with colorfulness term

Super-resolution

Feature/Perceptual loss
Deep feature matching objective

[Zhang et al. 2016]

[Gatys et al., 2016], [Johnson et al. 2016]
[Dosovitskiy and Brox. 2016]

Slide credit: Phillip Isola
“Perceptual Loss”


CNNs as a Perceptual Metric

(1) How well do “perceptual losses” describe perception?

CNNs as a Perceptual Metric

\[
\text{Perceptual Loss} = \arg \min_G \mathbb{E}_{(x,y)} \sum_{i=1}^{N} \lambda_i \frac{1}{M_i} \left\| F^{(i)}(G(x)) - F^{(i)}(y) \right\|_2^2
\]

The number of elements in the \((i)\)-th layer
What has a CNN Learned?

Perceptual Loss

\[
\arg \min_G \mathbb{E}_{(x,y)} \sum_{i=1}^{N} \lambda_i \frac{1}{M_i} \left\| F^{(i)}(G(x)) - F^{(i)}(y) \right\|_2^2
\]

The number of elements in the (i)-th layer

Slide credit: Richard Zhang
How Different are these Patches?

\[ D(\text{left}, \text{right}) \]

*The Unreasonable Effectiveness of Deep Features as a Perceptual Metric.* In *CVPR*, 2018.

Slide credit: Richard Zhang
Which patch is more similar to the middle?

Type 1

Type 2

Humans
L2/PSNR
SSIM/FSIMc

Deep Networks?
% agreement with human judges

Networks perform strongly across supervision signals and architectures

Fitting some data is important

VGG ("perceptual loss") correlates well

Slide credit: Richard Zhang
“Perceptual Loss”


Generated images

Universal loss?
Learning with Human Perception

Generated images

Human Annotation

Real vs. Fake

Real photos

[Zhu et al. 2014]
Generative Adversarial Network (GANs)

Real photos → Classifier → Real vs. Fake

Generated images

[Goodfellow, Pouget-Abadie, Mirza, Xu, Warde-Farley, Ozair, Courville, Bengio 2014]
Image synthesis from “noise”

\[ G : \mathcal{Z} \rightarrow \mathcal{X} \]

\[ z \sim p(z) \]

\[ x = G(z) \]
Image synthesis from "noise"

$z \sim p(z) \rightarrow G \rightarrow x = G(z)$

$G : \mathcal{Z} \rightarrow \mathcal{X}$

$z \sim p(z)$

$x = G(z)$
Image synthesis from “noise”

\[ z \sim p(z) \rightarrow G \rightarrow \text{Generator} \rightarrow x = G(z) \]

\[ G : \mathcal{Z} \rightarrow \mathcal{X} \]
\[ z \sim p(z) \]
\[ x = G(z) \]
$z$  
Random code  

$G$  
Generator  

$G(z)$  
fake image  

[Goodfellow et al. 2014]
A two-player game:

- $G$ tries to generate fake images that can fool $D$.
- $D$ tries to detect fake images.

[Goodfellow et al. 2014]
Learning objective (GANs)

\[
\min_G \max_D \mathbb{E}_z \left[ \log(1 - D(G(z))) \right] + \mathbb{E}_x \left[ \log D(x) \right]
\]

[Goodfellow et al. 2014]
Learning objective (GANs)

$$\min_G \max_D \mathbb{E}_z \left[ \log(1 - D(G(z))) \right] + \mathbb{E}_x \left[ \log D(x) \right]$$

[Goodfellow et al. 2014]
Learning objective (GANs)

$$\min_G \max_D \mathbb{E}_z [\log(1 - D(G(z)))] + \mathbb{E}_x [\log D(x)]$$

[Goodfellow et al. 2014]
GANs Training Breakdown

- From the discriminator $D$’s perspective:
  - binary classification: real vs. fake.
  - Nothing special: similar to 1 vs. 7 or cat vs. dog

$$\max_D \mathbb{E}[\log(1-D(\text{真实}))] + \mathbb{E}[\log D(\text{假的})]$$
GANs Training Breakdown

• From the discriminator D’s perspective:
  • binary classification: real vs. fake.
  • Nothing special: similar to 1 vs. 7 or cat vs. dog

\[
\max_D \mathbb{E}[\log(1 - D(\text{dog}))] + \mathbb{E}[\log D(\text{cat})]
\]

• From the generator G’s perspective:
  • Optimizing a loss that depends on a classifier D
  • We have done it before (Perceptual Loss)

\[
\min_G \mathbb{E}_z[\mathcal{L}_D(G(z))] \quad \min_G \mathbb{E}_{(x,y)}||F(G(x)) - F(y)||
\]

GAN loss for G \quad \text{Perceptual Loss for G}
GANs Training Breakdown

G tries to synthesize fake images that fool D

D tries to identify the fakes

- Training: iterate between training D and G with backprop.
- Global optimum when G reproduces data distribution.

[Goodfellow et al., 2014]
\[ p_g = p_{data} \] is the unique global minimizer of the GAN objective.

Proof

Optimal discriminator given fixed G

\[
C(G) = E_{x \sim p_{data}} \left[ \log D^*_{G}(x) \right] + E_{x \sim p_g} \left[ \log (1 - D^*_{G}(x)) \right]
\]

\[
= E_{x \sim p_{data}} \left[ \log \frac{p_{data}(x)}{p_{data}(x) + p_g(x)} \right] + E_{x \sim p_g} \left[ \log \frac{p_g(x)}{p_{data}(x) + p_g(x)} \right]
\]

\[
C(G') = -\log(4) + KL \left( p_{data} \left\| \frac{p_{data} + p_g}{2} \right\| \right) + KL \left( p_g \left\| \frac{p_{data} + p_g}{2} \right\| \right)
\]

\[
C(G') = -\log(4) + 2 \cdot JSD \left( p_{data} \parallel p_g \right)
\]

\[ \geq 0, \quad 0 \iff p_g = p_{data} \quad \square \]

KLD (Kullback–Leibler divergence): \[ KL(p \parallel q) = \int p(x) \log \frac{p(x)}{q(x)} dx \]

JSD (Jensen–Shannon divergence): \[ JSD(p \parallel q) = \frac{1}{2} KL(p \parallel \frac{p + q}{2}) + \frac{1}{2} KL(q \parallel \frac{p + q}{2}) \]
Generative Adversarial Network

Learner

Objective
\[ \min_G \max_D \mathbb{E}_z[\log(1 - D(G(z)))] + \mathbb{E}_x[\log D(x)] \]

Hypothesis space
Deep nets G and D

Optimizer
Alternating SGD on G and D

Critic
\[ D : \mathcal{X} \rightarrow [0, 1] \]

Sampler
\[ G : \mathcal{Z} \rightarrow \mathcal{X} \]
Generative Adversarial Network

\[ \text{Data} \rightarrow \text{Learner} \rightarrow G : \mathcal{Z} \rightarrow \mathcal{X} \]

Objective
\[ JSD (p_{data} \parallel p_g) \]
Hypothesis space
Deep net G
Optimizer
Adversarial game
What has driven GAN progress?

Ian Goodfellow @goodfellow_ian · Jan 14

![Images of face generation progression from 2014 to 2018](image-url)
\[ D^*_G(x) = \frac{p_{\text{data}}(x)}{p_{\text{data}}(x) + p_g(x)} \]

\[ \log D(G(z)) \rightarrow -\infty \]

from [Arjovsky, Chintala, Bottou, 2017]
Learning objective (GANs)

\[
\min_G \max_D \mathbb{E}_z [\log (1 - D(G(z)))] + \mathbb{E}_x [\log D(x)]
\]

[Goodfellow et al. 2014]
Learning objective (GANs variants)

\[
\min_G \max_{f_1,f_2} \mathbb{E}_z[f_1(G(z))] + \mathbb{E}_x[f_2(x)]
\]

EBGAN, WGAN, LSGAN, etc

[Goodfellow et al. 2014]
Other divergences?

from [Mohamed & Lakshminarayanan 2017]

Convenient choice

\[
\min_G \max_{f_1, f_2} \mathbb{E}_z[f_1(G(z))] + \mathbb{E}_x[f_2(x)]
\]

Different choices of \( f_1 \) and \( f_2 \) correspond to different divergence measures:

- Original GAN \( \rightarrow \) JSD
- Least-squares GAN \( \rightarrow \) Pearson chi-squared divergence

\[
\min_D V_{\text{LSGAN}}(D) = \frac{1}{2} \mathbb{E}_{x \sim p_{\text{data}}}(x) \left[ (D(x) - 1)^2 \right] + \frac{1}{2} \mathbb{E}_{z \sim p_z}(z) \left[ (D(G(z)) - 1)^2 \right]
\]

\[
\min_G V_{\text{LSGAN}}(G) = \frac{1}{2} \mathbb{E}_{z \sim p_z}(z) \left[ (D(G(z)) - 1)^2 \right].
\]
Other divergences?

\[ KL(p_{\text{data}} \| p_{\theta}) \quad \leftarrow \quad \mathbb{E}_{x \sim p_{\text{data}}} [\log p_{\theta}(x)] \]

\[ KL(p_{\theta} \| p_{\text{data}}) \quad \leftarrow \quad \text{Reverse KL — mode seeking, intractable} \]

\[ JS(p_{\text{data}}, p_{\theta}) \quad \leftarrow \quad \text{Jensen-Shannon, original GAN} \]

\[ W(p_{\text{data}}, p_{\theta}) = \inf_{\gamma \in \Pi(p_{\text{data}}, p_{\theta})} \mathbb{E}_{(x,y) \sim \gamma} [\|x - y\|] \quad \leftarrow \quad \text{Wasserstein} \]

Earth-Mover (EM) distance / Wasserstein distance
Maximum log likelihood, KL, and JSD

KLD (Kullback–Leibler divergence): \( \mathcal{KL}(p||q) = \int p(x) \log \frac{p(x)}{q(x)} \, dx \)

JSD (Jensen–Shannon divergence): \( \mathcal{JSD}(p \parallel q) = \frac{1}{2} \mathcal{KL}(p \parallel \frac{p+q}{2}) + \frac{1}{2} \mathcal{KL}(q \parallel \frac{p+q}{2}) \)

\[ \mathbb{E}_{x \sim p_{\text{data}}(x)} \left[ \log p_{\theta}(x) \right] = \int_x p_{\text{data}}(x) \log p_{\theta}(x) \, dx \]

\[ \mathcal{KL}(p_{\text{data}}(x) || p_{\theta}(x)) = \int_x p_{\text{data}}(x) \log \frac{p_{\text{data}}(x)}{p_{\theta}(x)} \, dx \]

\[ = \int_x p_{\text{data}}(x) \log p_{\text{data}}(x) \, dx - \int_x p_{\text{data}}(x) \log p_{\theta}(x) \, dx \]

Constant (independent of \( \theta \))

Maximize log likelihood = minimize KLD
Maximum log likelihood/KL vs. JSD

Data

Max likelihood / KL

Jensen-Shannon Divergence

[Theis et al. 2016]
Other divergences?

\[ KL(\rho_{data} \mid \rho_\theta) \leftarrow \mathbb{E}_{x \sim \rho_{data}}[\log \rho_\theta(x)] \]

\[ KL(\rho_\theta \mid \rho_{data}) \leftarrow \text{Reverse KL — mode seeking, intractable} \]

\[ JS(\rho_{data}, \rho_\theta) \leftarrow \text{Jensen-Shannon, original GAN} \]

\[ W(\rho_{data}, \rho_\theta) = \inf_{\gamma \in \Pi(\rho_{data}, \rho_\theta)} \mathbb{E}_{(x,y) \sim \gamma}[\|x - y\|] \leftarrow \text{Wasserstein} \]

Earth-Mover (EM) distance / Wasserstein distance
Wasserstein GAN

[Arjovsky, Chintala, Bottou 2017]

\[
\arg\min_G \max_{\|f\|_{L^1} \leq 1} \mathbb{E}_{z,x} \left[ -f(G(z)) \right] + f(x)
\]

Lipschitz continuity
\[
|f(x) - f(y)| \leq |x - y|
\]

wGAN GP [Gulrajani et al., 2018]:

\[
\arg\min_G \max_f \mathbb{E}_{z,x} \left[ -f(G(z)) \right] + f(x) + \lambda \mathbb{E}_{\hat{x} \sim \mathbb{P}_{x}} [(\|\nabla_{\hat{x}} f(\hat{x})\|_2 - 1)^2]
\]

Gradient penalty (GP)
from [Arjovsky, Chintala, Bottou, 2017]
To be continued...
Thank You!

16-726, Spring 2022
https://learning-image-synthesis.github.io/sp22/