Announcement

• HW1 winner: Riyaz Panjwani

Honorable mention: Harry Freeman
Announcement

• HW3 (due: 3/21/2022)
Style and Content, Texture Synthesis (part II)

Jun-Yan Zhu
16-726, Spring 2022

Many slides are from Taesung Park, Fujun Luan, etc.
Loss Functions
(Image-to-Image Translation)
Style and Content

Adversarial loss
\[ \mathbb{E}_x \log(1 - D_Y(G(x))) + \mathbb{E}_y \log D_Y(y) \]

\[ p(x) \rightarrow p(y) \text{ change style} \]

Cycle-consistency loss
\[ \mathbb{E}_x ||F(G(x)) - x||_1 \]

Bidirectional: preserve content

Separating Style and Content
[Tenenbaum and Freeman 1996]
**Style and Content**

Input image → **G** → Output image → **D** → Real (1) or fake (0)?

### Adversarial loss (change style)

\[
\mathbb{E}_x \log(1 - D_Y(G(x))) + \mathbb{E}_y \log D_Y(y)
\]

### L1 loss (preserve content in pixel space)

\[
\mathbb{E}_x \|G(x) - x\|_1
\]

SimGAN [Shrivastava et al., 2017]
Style and Content

Adversarial loss (change style)

$$\mathbb{E}_x \log(1 - D_Y(G(x))) + \mathbb{E}_y \log D_Y(y)$$

Feature loss (Preserve content in feature space)

$$\mathbb{E}_x \| F(G(x)) - F(x) \|$$

DTN [Taigman et al., 2017]
Perceptual/Feature Loss

(Conditional) GAN Loss

Cycle-Consistency Loss

Patch-wise Contrastive Loss
Style and Content

• **Style**: domain-specific features
  (horse vs. zebra)

• **Content**: features shared across two domains
Loss Functions
(Neural Style Transfer)
Neural Style Transfer

content image + style image = output result

[Gatys et al., 2016]
Style Reconstruction (Style Loss)

\[ |\text{Gram}(\hat{y}) - \text{Gram}(y)| \]

\[ \text{optimized output} \quad \quad \quad \text{style image} \]

Gram = Gram Matrix of a deep network’s features (e.g., ImageNet classifier)

Style Loss

\[ \arg \min_{\hat{y}} \sum_{j}^{M} \lambda_j \| \text{Gram}^{(j)}(\hat{y}) - \text{Gram}^{(j)}(y) \|^2 \]

[Gatys et al., 2016]
Computing Gram Matrix

Gram matrix:
- Cross Correlation of CNN features
- Invariant to the feature locations

\[ G_{ij} = \langle v_i, v_j \rangle \quad G = V^\top V \]

\[ Gram^{(j)}(x) = \frac{1}{C_j H_j W_j} \sum_{h=1}^{H_j} \sum_{w=1}^{W_j} \phi_j(x)_{h,w,c} \phi_j(x)_{h,w,c'} \]

h, w: pixel locations index

\[ \phi_j(x)_{h,w,c} \]

c: channel index

\[ \phi_j(x)_{h,w,c'} \]

H, W: height and width of feature map

C: the number of total channels
Content Reconstruction (Perceptual Loss)

\[
\hat{y} - F(\hat{y}) - F(x)
\]

F is a deep network (e.g., ImageNet classifier)

Content Loss

\[
\arg\min_{\hat{y}} \sum_{i} \lambda_{i} \left\| F^{(i)}(\hat{y}) - F^{(i)}(x) \right\|_{1}
\]

[14]

Gatys et al., 2016
Neural Style Transfer

\[
\arg \min_{\hat{y}} \mathcal{L}_{\text{style}}(\hat{y}, y) + \lambda \mathcal{L}_{\text{content}}(\hat{y}, x)
\]

\[
|\text{Gram}(\hat{y}) - \text{Gram}(y)|
\]

\[
+ |\text{F}(\hat{y}) - \text{F}(y)|
\]

\[\text{optimized output} \quad \text{style image} \]

\[\text{optimized output} \quad \text{content image}\]
Fast Neural Style Transfer

• Optimization-based method

\[
\arg\min_{\hat{y}} \mathcal{L}_{\text{style}}(\hat{y}, y) + \lambda \mathcal{L}_{\text{content}}(\hat{y}, x)
\]

• Feedforward network

\[
\arg\min_{G} \mathbb{E}_x \mathcal{L}_{\text{style}}(G(x), y) + \lambda \mathcal{L}_{\text{content}}(G(x), x)
\]

[Johnson et al., 2016]
Arbitrary Style Transfer with AdaIN

- Feedforward network with any style

\[ \arg \min_G \mathbb{E}_{x,y} \mathcal{L}_{\text{style}}(G(x, y), y) + \lambda \mathcal{L}_{\text{content}}(G(x, y), x) \]
Arbitrary Style Transfer with AdaIN

[Huang et al., 2017]
Style and Content

• **Style**: the style (color, texture, etc.) of a single painting

• **Content**: the layout and semantics of a real photo
Disentangled Latent Space
Disentangling the Latent Space

- **UNIT**
  - A single *shared, domain-invariant* latent space \( \mathcal{Z} \)

\[ \mathcal{X}_1 \quad \mathcal{Z} \quad \mathcal{X}_2 \]

[Liu et al., 2017]
Disentangling the Latent Space

- Multimodal UNIT (MUNIT)
  - A **content** space $\mathcal{C}$ that is shared, domain-invariant
  - Two **style** spaces $\mathcal{S}_1, \mathcal{S}_2$ that are unshared, domain-specific
Style and Content

• **Style**: variations within the same domain (different colors, textures, etc.)
• **Content**: features shared across two domains
Are and from the same image?
Are and from the same image?

Answer: No

Swapping Autoencoder [Park et al., 2020]
Are and from the same image?

Answer: Yes

Swapping Autoencoder [Park et al., 2020]
Are and from the same image?

Patch co-occurrence discriminator

Answer: ...?
Auto-encode

\[ E \] structure code

texture code

\[ G \]

Swapping Autoencoder [Park et al., 2020]
Swapping Autoencoder [Park et al., 2020]
Auto-encode

Swap

Reconstruction

$E$

$G$

Reference patches

Patch co-occurrence discriminator $D_{patch}$

Real/fake?
Style and Content

- **Style**: variations within the same domain (different colors, textures, etc.)
- **Content**: the layout and semantics of a single photo
Neural Style Transfer

vs.

Image-to-Image Translation
Photo $\rightarrow$ Van Gogh

horse $\rightarrow$ zebra
Photo Style Transfer
Deep Photo Style Transfer

Local color transfer? (hard to transfer texture)

[Luan et al., 2017]
Make [image] look like [image]
Make Histogram Matching look like
Make look like

Reinhard et al. [2001]
Make look like

Pitie et al. [2005]
Make

Photoshop
Match Color

look like
Make look like

Gatys et al. [2016]
Make look like

Our method
Ours
Motivation

The neural style algorithm...

– Works well for Paintings!

– What about Photos?
Motivation

• So we tried it on photos:

Input

Style

Result
Fixing Distortion

Local affine color transform for each patch [Levin et al. 2006]

\[
\begin{pmatrix}
  r_{out} \\
  g_{out} \\
  b_{out}
\end{pmatrix}
= A_{3\times3} \begin{pmatrix}
  r_{in} \\
  g_{in} \\
  b_{in}
\end{pmatrix} + B_{3\times1}
\]

See more technical details on Wednesday’s paper presentation
Single Image Translation

Domain = \{\text{patches of a single image}\}
Internal contrastive loss is well-suited for single image translation. Also see InGAN (Shocher et al., ICCV’19), SinGAN (Shaham et al., ICCV’19)
Internal contrastive loss is well-suited for single image translation. Also see InGAN (Shocher et al., ICCV’19), SinGAN (Shaham et al., ICCV’19)
Single Image Translation

Claude Monet’s painting

Internal contrastive loss is well-suited for single image translation. Also see InGAN (Shocher et al., ICCV’19), SinGAN (Shaham et al., ICCV’19)
Painting
Reference

STROTSS (Kolkin et al., CVPR’19)

Deep Image Analogy’s extension
Reference

Painting

WCT² (Yoo et al., ICCV’19)

Photo style transfer’s extension
Reference

Painting

Our translation result
Image-to-Image Translation for Texture Synthesis
Texture Synthesis by Conditional GANs

Texture Synthesis by Conditional GANs

Output

Input

Output
Texture Synthesis by Conditional GANs

Input | Output 1 | Output 2 | Input | Output 1 | Output 2
Random crops of the large result
Style Transfer vs. Image-to-Image Translation

• Data (how to define Style)
  – A single image? A collection of images?

• Applications
  – Photo -> Painting (Neural Style Transfer, Image-to-Image Translation)
  – Photo -> Photo (Image-to-Image Translation, Photo Style Transfer (Color))
  – Painting -> Photo (Image-to-Image Translation, Deep Image Analogy)

• Algorithms:
  – Patch-based method (i.e., correspondence between output and input)
  – Optimization-based method
  – Feed-forward network

• Loss functions
  – Style Loss: GAN loss, Gram matrix loss
  – Content Loss: Perceptual Loss (L2 reconstruction loss), identity loss, conditional GAN Loss, Cycle-consistency loss, Contrastive Loss (InfoNCE)
Style Transfer + Poisson Blending
Motivation: Image compositing on paintings
Poisson blending
Ours

Deep Painterly Harmonization
[Luan et al., 2018]
Intuition 3: Two-pass framework

- Two-pass harmonzation is more robust than one-pass version
More results
Deep Painterly Harmonization [Luan et al., 2018]
More results
Deep Painterly Harmonization [Luan et al., 2018]
More results

Deep Painterly Harmonization [Luan et al., 2018]