Deep Learning for Computer Vision

Deep Generative Models across Multiple Domains

Vineeth N Balasubramanian

Department of Computer Science and Engineering
Indian Institute of Technology, Hyderabad

Review

Questions

- Minibatch Standard Deviation is used in Progressive GAN. Why is this useful?

Review

Questions

- Minibatch Standard Deviation is used in Progressive GAN. Why is this useful?
  If the generated images do not have the same diversity as the real images, this value will be different and therefore will be penalized by the discriminator

Review

Questions

- Minibatch Standard Deviation is used in Progressive GAN. Why is this useful?
  If the generated images do not have the same diversity as the real images, this value will be different and therefore will be penalized by the discriminator
- Orthogonal Regularization of weights is used in BigGAN. Why is this useful?

Review

Questions

- Minibatch Standard Deviation is used in Progressive GAN. Why is this useful?
  If the generated images do not have the same diversity as the real images, this value will be different and therefore will be penalized by the discriminator
- Orthogonal Regularization of weights is used in BigGAN. Why is this useful?
  Multiplication by an orthogonal matrix leaves the norm of the original matrix unchanged. Why is this useful?

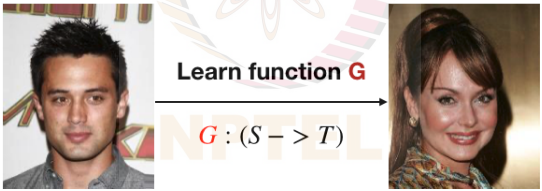
Review

Questions

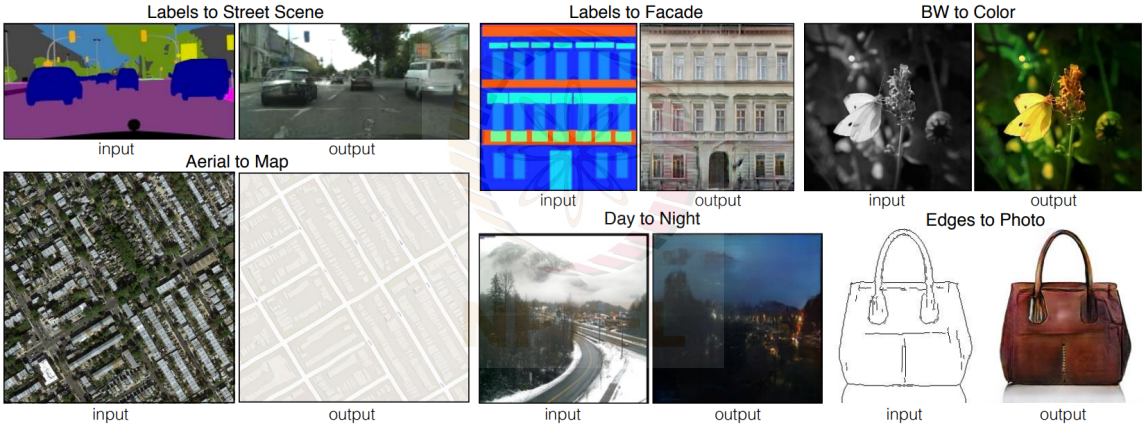
- Minibatch Standard Deviation is used in Progressive GAN. Why is this useful?
  If the generated images do not have the same diversity as the real images, this value will be different and therefore will be penalized by the discriminator
- Orthogonal Regularization of weights is used in BigGAN. Why is this useful?
  Multiplication by an orthogonal matrix leaves the norm of the original matrix unchanged. Why is this useful? Recall weight initialization and batch normalization. It is useful to maintain the same norm across all layers!

Domain Translation

- Given an image from a source domain, generate an image in a target domain
- Learn a function G for the mapping $G : (S- > T)$
- Examples: Male-to-female, sketches-to-photos, summer-to-winter, etc



Domain Translation: Examples



Credit: Isola et al, Image-to-Image Translation with Conditional Adversarial Networks, CVPR 2017

Domain Translation: Challenges

- **Paired training (supervised)**
  - Pix2Pix[1]
- **Unpaired training (unsupervised)**
  - Cycle-GAN[2]
- **Multi-modal generation**
  - UNIT-GAN[3] and MUNIT-GAN[4]

[1] Isola et al, Image-to-Image Translation with Conditional Adversarial Networks, CVPR 2017
[2] Zhu et al, Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks, ICCV 2017
[3] Liu et al, Unsupervised Image-to-Image Translation Networks, NeurIPS 2017
[4] Huang et al, Multimodal Unsupervised Image-to-Image Translation, ECCV 2018

Pix2Pix[5]

Redefines image-to-image translation tasks to **predicting pixels from pixels** and provides a common framework to perform such tasks

[5] Isola et al, Image-to-Image Translation with Conditional Adversarial Networks, CVPR 2017

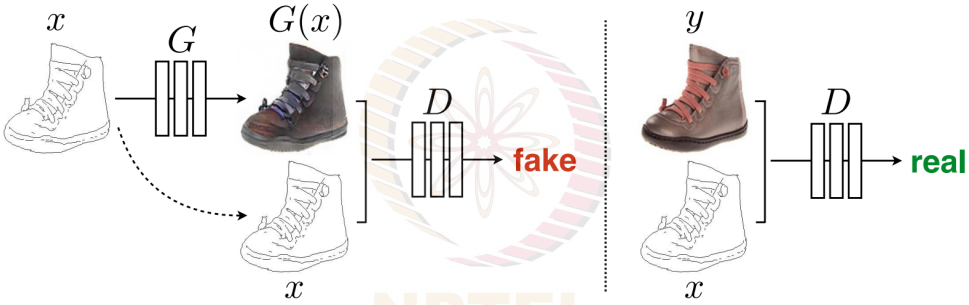
Pix2Pix[5]

Normal GAN objective:

$$\mathcal{L}_{GAN}(G, D) = \mathbb{E}_x[\log D(x)] + \mathbb{E}_z[1 - \log D(G(z))]$$

[5] Isola et al, Image-to-Image Translation with Conditional Adversarial Networks, CVPR 2017

Pix2Pix[5]



Pix2Pix conditional GAN objective:

$$\mathcal{L}_{cGAN}(G, D) = \mathbb{E}_{x,y}[\log D(x, y)] + \mathbb{E}_{x,z}[\log(1 - D(x, G(x, z))]$$

[5] Isola et al, Image-to-Image Translation with Conditional Adversarial Networks, CVPR 2017

Pix2Pix: Final Objective

- Also uses L1 loss to force generator G to create images close to ground truth:

$$\mathcal{L}_{L1}(G) = \mathbb{E}_{x,y,z}[||y - G(x,z)||_1]$$

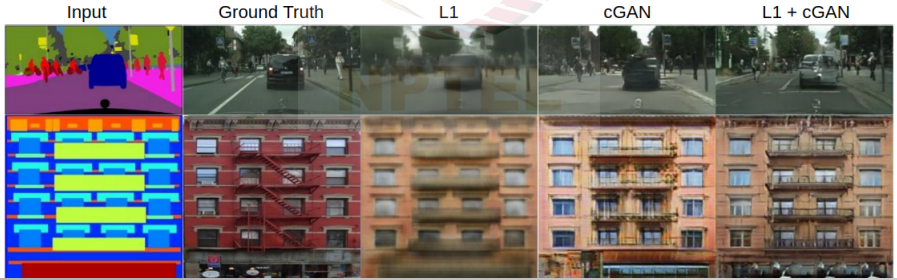
Pix2Pix: Final Objective

- Also uses L1 loss to force generator G to create images close to ground truth:

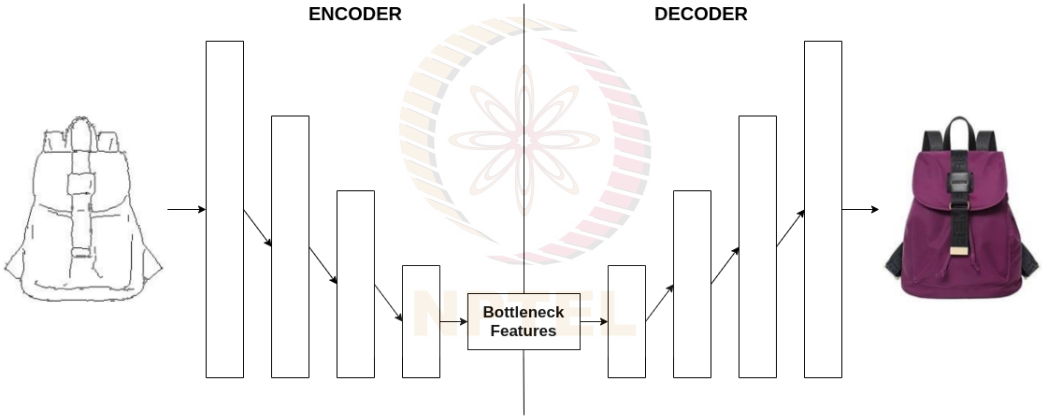
$$\mathcal{L}_{L1}(G) = \mathbb{E}_{x,y,z}[||y - G(x,z)||_1]$$

- Final Pix2Pix GAN objective:

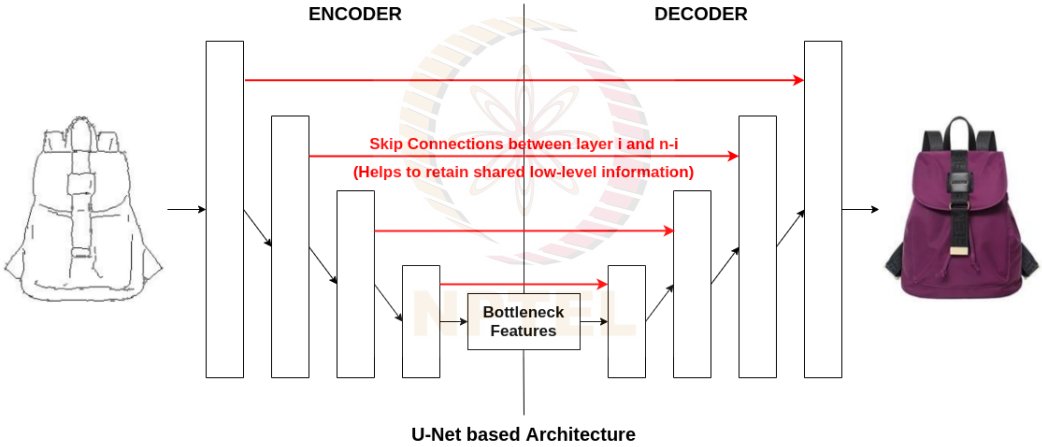
$$G^* = \arg\min_G \max_D \mathcal{L}_{cGAN}(G,D) + \lambda \mathcal{L}_{L1}(G)$$



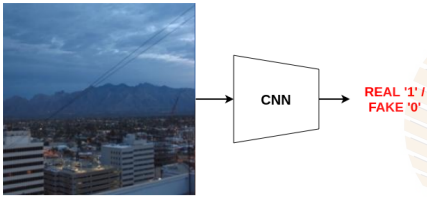
Pix2Pix: Generator Architecture



Pix2Pix: Generator Architecture



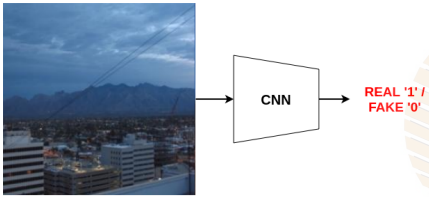
Pix2Pix: PatchGAN Discriminator



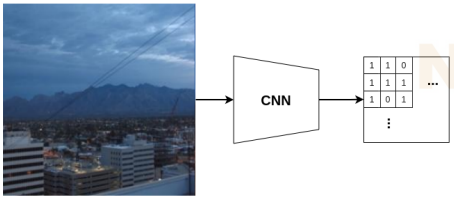
Normal Discriminator Approach

- L1 (or L2) loss ensures a crispness of low-frequency components of generated images

Pix2Pix: PatchGAN Discriminator



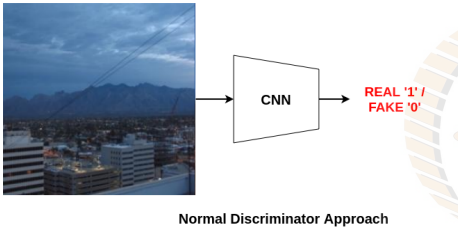
Normal Discriminator Approach



PatchGAN Discriminator Approach

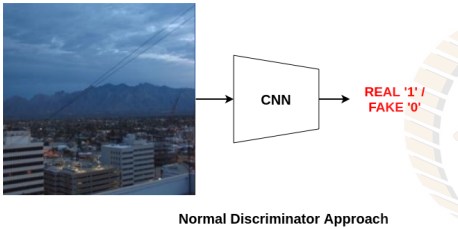
- L1 (or L2) loss ensures a crispness of low-frequency components of generated images
- To encourage high-frequency crispness, enforce losses at an $N \times N$ patch level $\rightarrow$ PatchGAN

Pix2Pix: PatchGAN Discriminator



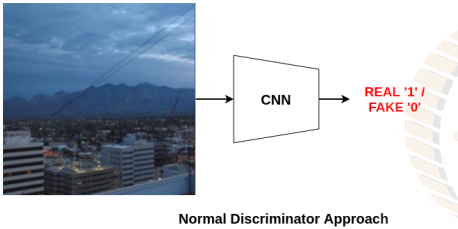
- L1 (or L2) loss ensures a crispness of low-frequency components of generated images
- To encourage high-frequency crispness, enforce losses at an $N \times N$ patch level $\rightarrow$ PatchGAN

Pix2Pix: PatchGAN Discriminator



- L1 (or L2) loss ensures a crispness of low-frequency components of generated images
- To encourage high-frequency crispness, enforce losses at an $N \times N$ patch level $\rightarrow$ PatchGAN

Pix2Pix: PatchGAN Discriminator



- L1 (or L2) loss ensures a crispness of low-frequency components of generated images
- To encourage high-frequency crispness, enforce losses at an $N \times N$ patch level $\rightarrow$ PatchGAN
- PatchGAN: a form of texture/style-aware generation

CycleGAN: Unsupervised Training[6]

- Paired data from different domains always difficult to collect

[6] Zhu et al, Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks, ICCV 2017

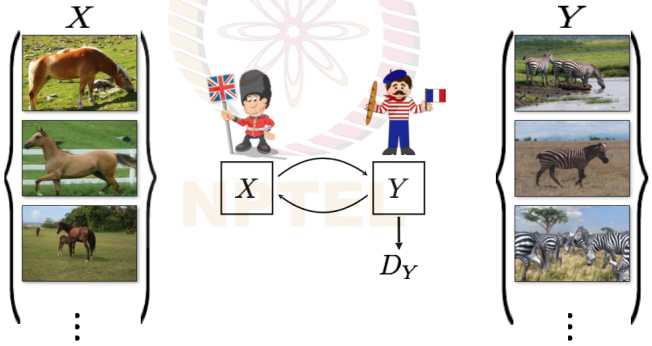
CycleGAN: Unsupervised Training[6]

- Paired data from different domains always difficult to collect
- Large amounts of unpaired data available, but difficult to learn domain-conditional distributions from such data
- Infinite possible translations for a given source sample!
- Solution: **CycleGAN**
  - Use two generators G and F which are inverse of each other
  - Use **cyclic consistency** where output from target domain should map back to source domain
  - Use adversarial training for generators and discriminators

[6] Zhu et al, Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks, ICCV 2017

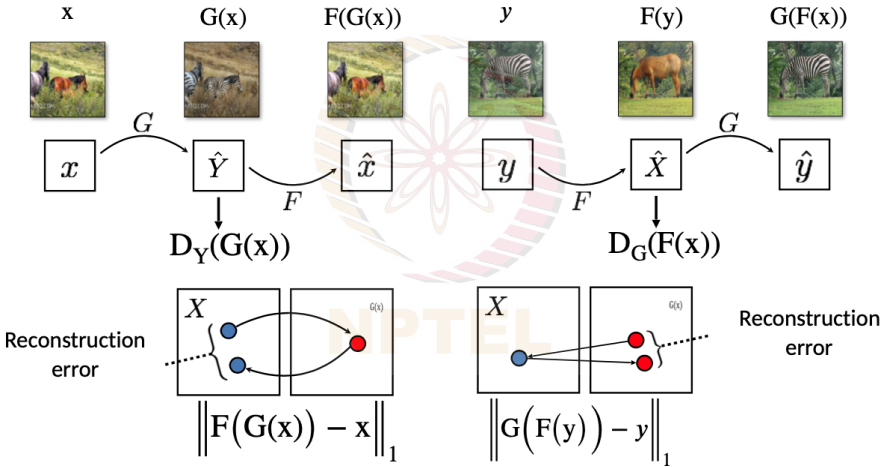
CycleGAN: Cyclic Consistency[7]

- Concept from machine translation where a phrase translated from English to French should translate from French back to English and be identical to the original phrase
- Reverse process should also be true



[7] Zhu et al, Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks, ICCV 2017

CycleGAN: Full Training[8]



[8] Zhu et al, Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks, ICCV 2017

CycleGAN: Adversarial loss

- Loss for domain $(X \rightarrow Y)$:

$$\mathcal{L}_{Adv}^{Y}(G_{XY}, D_Y) = \mathop{\mathbb{E}}_{y \sim p_{data}(y)} \Big[\log D_Y(y) \Big] + \mathop{\mathbb{E}}_{x \sim p_{data}(x)} \Big[\log(1 - D_Y(G_{XY}(x))) \Big]$$

CycleGAN: Adversarial loss

- Loss for domain $(X \to Y)$:

$$\mathcal{L}^{Y}_{Adv}(G_{XY}, D_Y) = \mathop{\mathbb{E}}_{y \sim p_{data}(y)} \Big[\log D_Y(y) \Big] + \mathop{\mathbb{E}}_{x \sim p_{data}(x)} \Big[\log(1 - D_Y(G_{XY}(x))) \Big]$$

- Loss for domain $(Y \to X)$:

$$\mathcal{L}^{Y}_{Adv}(G_{YX}, D_X) = \mathop{\mathbb{E}}_{y \sim p_{data}(x)} \Big[\log D_X(x) \Big] + \mathop{\mathbb{E}}_{y \sim p_{data}(y)} \Big[\log(1 - D_X(G_{YX}(y))) \Big]$$

CycleGAN: Adversarial loss

- Loss for domain $(X \to Y)$:

$$\mathcal{L}^{Y}_{Adv}(G_{XY}, D_Y) = \mathop{\mathbb{E}}_{y \sim p_{data}(y)} \Big[\log D_Y(y) \Big] + \mathop{\mathbb{E}}_{x \sim p_{data}(x)} \Big[\log(1 - D_Y(G_{XY}(x))) \Big]$$

- Loss for domain $(Y \to X)$:

$$\mathcal{L}^{Y}_{Adv}(G_{YX}, D_X) = \mathop{\mathbb{E}}_{y \sim p_{data}(x)} \Big[\log D_X(x) \Big] + \mathop{\mathbb{E}}_{y \sim p_{data}(y)} \Big[\log(1 - D_X(G_{YX}(y))) \Big]$$

- Cyclic loss for domain $(X \to Y)$:

$$\mathcal{L}^{X}_{cyc} = \mathop{\mathbb{E}}_{x \sim p_{data}(x)} ||G_{YX}(G_{XY}(x)) - x||_1 \; = \mathop{\mathbb{E}}_{x \sim p_{data}(x)} ||x' - x||_1.$$

CycleGAN: Adversarial loss

- Loss for domain $(X \to Y)$:

$$\mathcal{L}^{Y}_{Adv}(G_{XY}, D_Y) = \mathop{\mathbb{E}}_{y \sim p_{data}(y)} \Big[\log D_Y(y) \Big] + \mathop{\mathbb{E}}_{x \sim p_{data}(x)} \Big[\log(1 - D_Y(G_{XY}(x))) \Big]$$

- Loss for domain $(Y \to X)$:

$$\mathcal{L}^{Y}_{Adv}(G_{YX}, D_X) = \mathop{\mathbb{E}}_{y \sim p_{data}(x)} \Big[\log D_X(x) \Big] + \mathop{\mathbb{E}}_{y \sim p_{data}(y)} \Big[\log(1 - D_X(G_{YX}(y))) \Big]$$

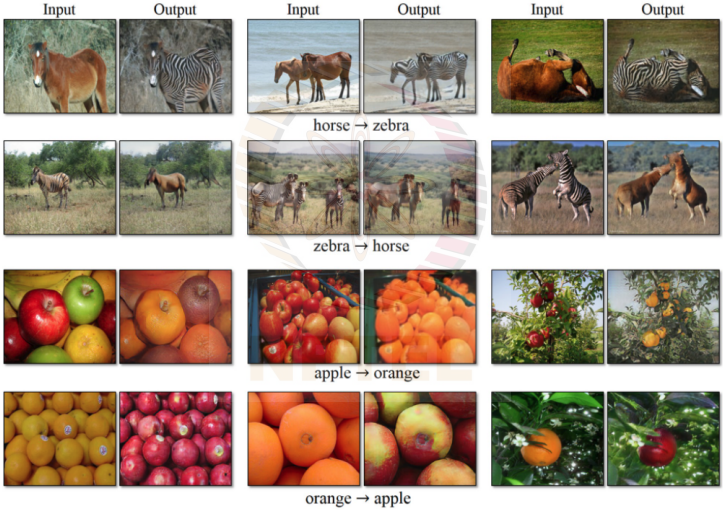
- Cyclic loss for domain $(X \to Y)$:

$$\mathcal{L}^{X}_{cyc} = \mathop{\mathbb{E}}_{x \sim p_{data}(x)} ||G_{YX}(G_{XY}(x)) - x||_1 = \mathop{\mathbb{E}}_{x \sim p_{data}(x)} ||x' - x||_1.$$

- Cyclic loss for domain $(Y \to X)$:

$$\mathcal{L}^{Y}_{cyc} = \mathop{\mathbb{E}}_{y \sim p_{data}(y)} ||G_{XY}(G_{YX}(y)) - y||_1 = \mathop{\mathbb{E}}_{x \sim p_{data}(x)} ||y' - y||_1.$$

Cycle-GAN: Results[9]



[9] Zhu et al, Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks, ICCV 2017

Cycle-GAN: More Results[10]

[10] Zhu et al, Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks, ICCV 2017

Multimodal Translation: Mode Collapse

- For a given source image, multiple possible translations could be present

Multimodal Translation: Mode Collapse

- For a given source image, multiple possible translations could be present
- Cycle-GAN suffers from **mode collapse**, where model does not produce diverse images

Multimodal Translation: Mode Collapse

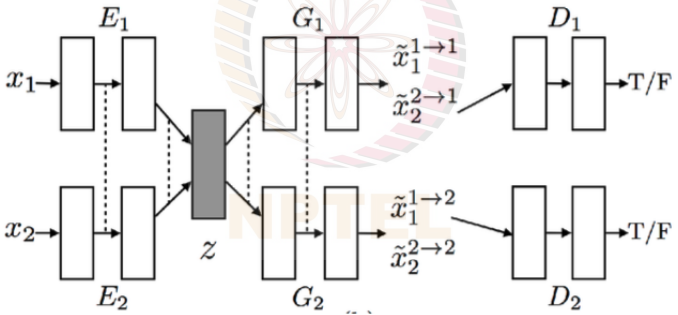
- For a given source image, multiple possible translations could be present
- Cycle-GAN suffers from **mode collapse**, where model does not produce diverse images
- **Solution:** Embed latent spaces within GAN network; latent space can be varied to obtain diverse images

(a) edges ↔ shoes (b) edges ↔ handbags

GT = Ground Truth

UNIT-GAN: Multimodal Translation[11]

- Use a VAE-GAN framework to learn latent spaces and domain translation
- Introduce cyclic consistency of latent spaces along with domains



[11] Liu et al, Unsupervised Image-to-Image Translation Networks, NeurIPS 2017

UNIT-GAN: Adversarial Loss

- VAE loss ($x_1 \rightarrow x_2$):

$$\mathcal{L}_{VAE_1}(E_1, G_1) = \lambda_1 KL\Big[q_1(z_1|x_1||p_\eta(z))\Big] - \lambda_2 \mathop{\mathbb{E}}_{z_1 \sim q_1(z_1|x_1)}\Big[\log p_{G_1}(x_1|z_1)\Big]$$

- GAN loss ($x_1 \rightarrow x_2$)

$$\mathcal{L}_{GAN_1}(E_1, G_1, D_1) = \lambda_0 \mathop{\mathbb{E}}_{x_1 \sim p_{x_1}}\Big[\log D_1(x_1)\Big] + \lambda_0 \mathop{\mathbb{E}}_{z_2 \sim q_2(z_2|x_2)}\Big[\log(1 - D_1(G_1(z_2)))\Big]$$

UNIT-GAN: Adversarial Loss

- VAE loss $(x_1 \rightarrow x_2)$:

$$\mathcal{L}_{VAE_1}(E_1, G_1) = \lambda_1 KL\Big[q_1(z_1|x_1||p_\eta(z))\Big] - \lambda_2 \mathop{\mathbb{E}}_{z_1 \sim q_1(z_1|x_1)}\Big[\log p_{G_1}(x_1|z_1)\Big]$$

- GAN loss $(x_1 \rightarrow x_2)$

$$\mathcal{L}_{GAN_1}(E_1, G_1, D_1) = \lambda_0 \mathop{\mathbb{E}}_{x_1 \sim p_{x_1}}\Big[\log D_1(x_1)\Big] + \lambda_0 \mathop{\mathbb{E}}_{z_2 \sim q_2(z_2|x_2)}\Big[\log(1 - D_1(G_1(z_2)))\Big]$$

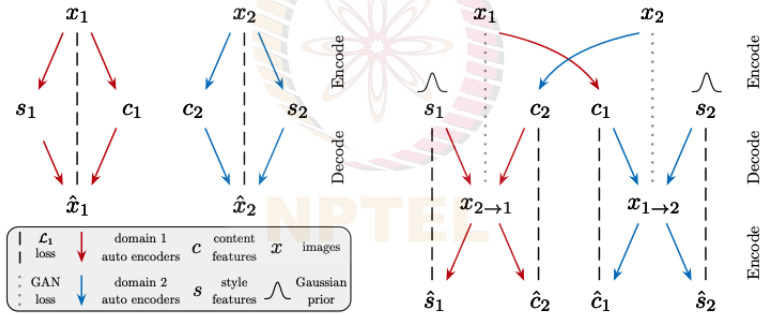
- Cyclic loss for $(x_1 \rightarrow x_2)$:

$$\mathcal{L}_{cc_1} = \lambda_3 KL\Big[q_1(z_1|x_1||p_\eta(z))\Big] + \lambda_3 KL\Big[q_2(z_2|x_1^{1\rightarrow 2}||p_\eta(z))\Big] - \lambda_4 \mathop{\mathbb{E}}_{z_2 \sim q_2(z_2|x_1^{1\rightarrow 2})}\Big[\log p_{G1}(x_1|z_2)\Big]$$

- Losses for $(x_2 \rightarrow x_1)$ defined similarly

MUNIT-GAN: Multimodal Translation[12]

- Divide image data into **content space** and domain-specific **style space**
- **Style encoder** learns specific style of each domain
- Use a within-domain auto-encoder framework and a cross-domain framework



(a) Within-domain reconstruction

(b) Cross-domain translation

[12] Huang et al, Multimodal Unsupervised Image-to-Image Translation, ECCV 2018

MUNIT-GAN: Training Objective

- **Image reconstruction:**

$$\mathcal{L}_{recon}^{x_1} = \mathop{\mathbb{E}}_{x_1 \sim p(x_1)} ||G_1(E_1^c(x_1)) - x_1||_1$$

MUNIT-GAN: Training Objective

- **Image reconstruction:**

$$\mathcal{L}_{recon}^{x_1} = \mathop{\mathbb{E}}_{x_1 \sim p(x_1)} ||G_1(E_1^c(x_1)) - x_1||_1$$

- **Latent reconstruction:**

$$\mathcal{L}_{recon}^{c_1} = \mathop{\mathbb{E}}_{c_1 \sim p(c_1), s_1 \sim q(s_1)} ||E_2^c(G_2(c_1, s_2)) - c_1||_1$$

$$\mathcal{L}_{recon}^{s_2} = \mathop{\mathbb{E}}_{c_1 \sim p(c_1), s_1 \sim q(s_1)} ||E_2^s(G_2(c_1, s_2)) - s_2||_1$$

where $q(s_2)$ is prior $N(0, I)$, $p(c_1)$ is given by $c_1 = E_1^c(x_1)$ and $x_1 \sim p(c_1)$, domains given by x_1 and x_2

MUNIT-GAN: Training Objective

- **Image reconstruction:**

$$\mathcal{L}_{recon}^{x_1} = \mathop{\mathbb{E}}_{x_1 \sim p(x_1)} ||G_1(E_1^c(x_1)) - x_1||_1$$

- **Latent reconstruction:**

$$\mathcal{L}_{recon}^{c_1} = \mathop{\mathbb{E}}_{c_1 \sim p(c_1), s_1 \sim q(s_1)} ||E_2^c(G_2(c_1, s_2)) - c_1||_1$$

$$\mathcal{L}_{recon}^{s_2} = \mathop{\mathbb{E}}_{c_1 \sim p(c_1), s_1 \sim q(s_1)} ||E_2^s(G_2(c_1, s_2)) - s_2||_1$$

where $q(s_2)$ is prior $N(0, I)$, $p(c_1)$ is given by $c_1 = E_1^c(x_1)$ and $x_1 \sim p(c_1)$, domains given by x_1 and x_2

- Other losses $\mathcal{L}_{recon}^{x_2}$, $\mathcal{L}_{recon}^{c_2}$, and $\mathcal{L}_{recon}^{s_1}$ defined similarly

MUNIT-GAN: Adversarial Loss

- Loss for domain $(x_1 \rightarrow x_2)$:

$$\mathcal{L}_{GAN}^{x_2} = \mathop{\mathbb{E}}_{c_1 \sim p(c_1), s_2 \sim q(s_2)} \Big[\log D_2(x_2) \Big] + \mathop{\mathbb{E}}_{x_2 \sim p(x_2)} \Big[\log(1 - D_2(G_2(c_1, s_1))) \Big]$$

MUNIT-GAN: Adversarial Loss

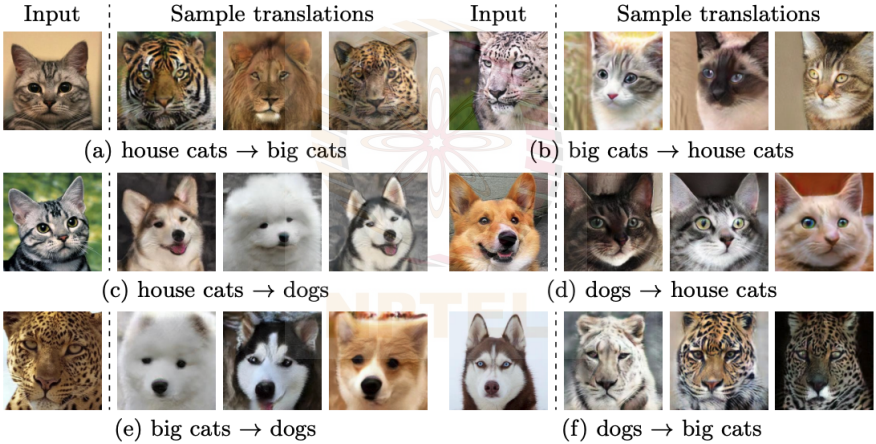
- Loss for domain $(x_1 \rightarrow x_2)$:

$$\mathcal{L}_{GAN}^{x_2} = \mathop{\mathbb{E}}_{c_1 \sim p(c_1), s_2 \sim q(s_2)} \Big[\log D_2(x_2) \Big] + \mathop{\mathbb{E}}_{x_2 \sim p(x_2)} \Big[\log(1 - D_2(G_2(c_1, s_1))) \Big]$$

- Loss for domain $(x_2 \rightarrow x_1)$:

$$\mathcal{L}_{GAN}^{x_1} = \mathop{\mathbb{E}}_{c_2 \sim p(c_2), s_1 \sim q(s_1)} \Big[\log D_1(x_1) \Big] + \mathop{\mathbb{E}}_{x_1 \sim p(x_1)} \Big[\log(1 - D_1(G_1(c_2, s_2))) \Big]$$

MUNIT-GAN: Results[13]



[13] Huang et al, Multimodal Unsupervised Image-to-Image Translation, ECCV 2018

MUNIT-GAN: Results[14]

(a) edges → shoes (b) big cats → house cats

[14] Huang et al, Multimodal Unsupervised Image-to-Image Translation, ECCV 2018

Homework

Readings

- Pix2Pix: https://phillipi.github.io/pix2pix/
- CycleGAN: https://junyanz.github.io/CycleGAN/
- UNIT: https://github.com/mingyuliutw/UNIT
- MUNIT: https://github.com/NVlabs/MUNIT
- List of all image-image translation work: https://github.com/lzhbrian/image-to-image-papers

References


Ming-Yu Liu, Thomas Breuel, and Jan Kautz. "Unsupervised image-to-image translation networks". In: *Advances in neural information processing systems*. 2017, pp. 700–708.


Jun-Yan Zhu et al. "Unpaired image-to-image translation using cycle-consistent adversarial networks". In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 2223–2232.


Jun-Yan Zhu et al. "Unpaired image-to-image translation using cycle-consistent adversarial networks". In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 2223–2232.


Xun Huang et al. "Multimodal unsupervised image-to-image translation". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018, pp. 172–189.