Enabling Deep Hierarchical Image-to-Image Translation by Transferring from GANs

Jimut Bahan Pal [1]
Team name: Zero1 (22D1594)

Under the Guidance Of

P. Balamurugan [2]

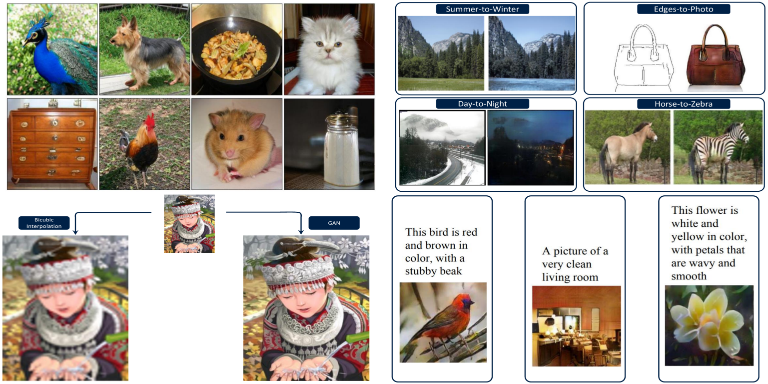
[1] Centre for Machine Intelligence and Data Science
[2] Industrial Engineering and Operations Research
Indian Institute of Technology, Bombay

IE643 Course Project (2022)

Table of Contents

1. Motivation
2. What are GANs?
3. Generator model
4. Discriminator model
5. Generator Objective
6. Discriminator Objective
7. Combined objective function
8. Problem in gradient
9. Modified objective
10. Algorithm for training GANs
11. Theoritical Analysis
12. DeepI2I: Summary
13. References
14. Acknowledgements

Motivation for Generating images[1]



[1] https://jonathan-hui.medium.com/gan-whats-generative-adversarial-networks-and-its-application-f39ed278ef09

Motivation for Generating images

Yann LeCun described GANs as **"the most interesting idea in the last 10 years in Machine Learning"**.

Motivation for Generating images

Table of Contents

1. Motivation
2. What are GANs?
3. Generator model
4. Discriminator model
5. Generator Objective
6. Discriminator Objective
7. Combined objective function
8. Problem in gradient
9. Modified objective
10. Algorithm for training GANs
11. Theoritical Analysis
12. DeepI2I: Summary
13. References
14. Acknowledgements

What are GANs?

- The training data comes from some underlying complex high-dimensional distribution $p_{data}(\mathbf{x})$.
- New data can be generated by sampling from this distribution using a generator $p_G(\mathbf{x})$.
- GANs overcome this problem by sampling from a simple distribution, and then learn a complex distribution to generate training data.
- The complex transformation is a deep neural network.

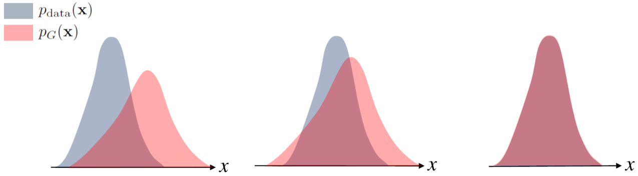
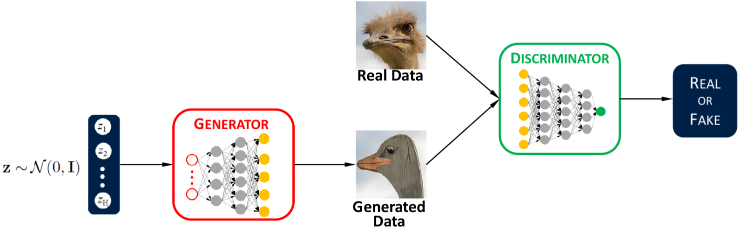


Figure 2: Generator tries to learn the underlying distribution.

What are GANs?

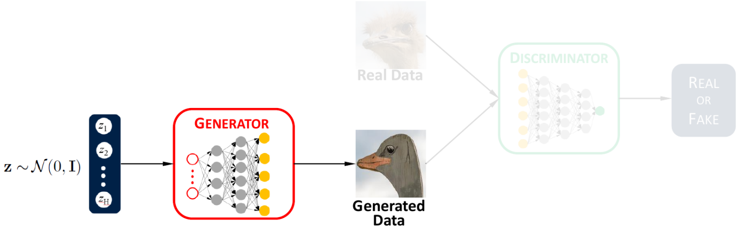


- Training is done using a two player game which comprise of a **generator** and a **discriminator**.
- Generator produces the images that appear to be real.
- The discriminator tries to detect if an image is real or fake.

Table of Contents

1 Motivation
2 What are GANs?
3 Generator model
4 Discriminator model
5 Generator Objective
6 Discriminator Objective
7 Combined objective function
8 Problem in gradient
9 Modified objective
10 Algorithm for training GANs
11 Theoritical Analysis
12 DeepI2I: Summary
13 References
14 Acknowledgements

Generator model

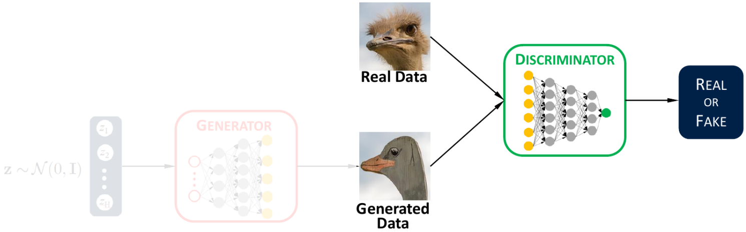


- $G_{\boldsymbol{\theta}}(\mathbf{z})$, where $\mathbf{z}$ is the input and $\boldsymbol{\theta}$ are the parameters of the model.
- Input: Noise vector $\mathbf{z} \sim p_{\mathbf{z}}(\mathbf{z})$ which can be $\mathcal{N}(0, \mathbf{I})$
- $G_{\boldsymbol{\theta}}$ is the neural network model.

Table of Contents

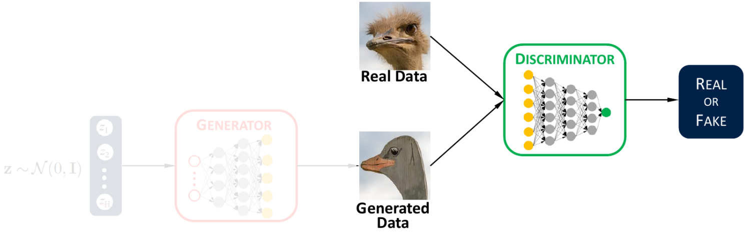
1 Motivation
2 What are GANs?
3 Generator model
4 Discriminator model
5 Generator Objective
6 Discriminator Objective
7 Combined objective function
8 Problem in gradient
9 Modified objective
10 Algorithm for training GANs
11 Theoritical Analysis
12 DeepI2I: Summary
13 References
14 Acknowledgements

Discriminator model



- $D_{\varphi}(\mathbf{x})$, where $\mathbf{x}$ is the input to the discriminator and φ are the parameters.
- Input can come from data or generator:
  - $\mathbf{x}$ if coming from the data.
  - $G_{\boldsymbol{\theta}}(\mathbf{z})$ if coming from the generator.
- D_{φ} is a neural network model.

Discriminator model

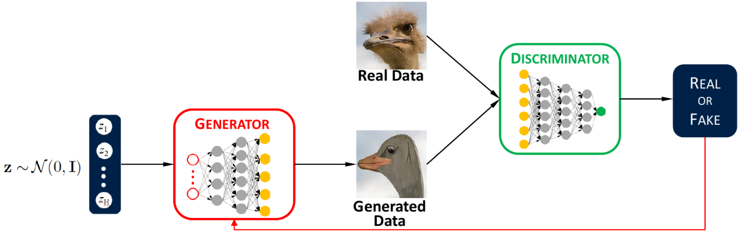


- The discriminator $D_\varphi(\mathbf{x})$ outputs a score between 0 and 1.
- This is a probability of an image being real or fake.
- It is 0 if the image is fake and 1 if it is real.

Table of Contents

1. Motivation
2. What are GANs?
3. Generator model
4. Discriminator model
5. Generator Objective
6. Discriminator Objective
7. Combined objective function
8. Problem in gradient
9. Modified objective
10. Algorithm for training GANs
11. Theoritical Analysis
12. DeepI2I: Summary
13. References
14. Acknowledgements

Generator Objective

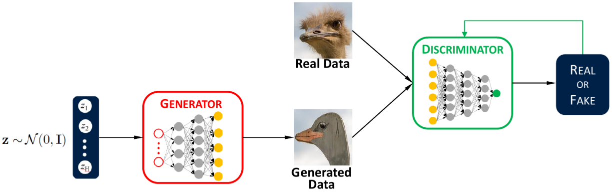


- The generator wants its output $G_{\boldsymbol{\theta}}(\mathbf{z})$ to be classified as real. Therefore, it wants to: $\max\limits_{\boldsymbol{\theta}} \log D_{\boldsymbol{\varphi}}(G_{\boldsymbol{\theta}}(\mathbf{z}))$ *or* $\min\limits_{\boldsymbol{\theta}} \log(1 - D_{\boldsymbol{\varphi}}(G_{\boldsymbol{\theta}}(\mathbf{z}))$
- We want the generator to do this task for all possible values of $\mathbf{z}$ sampled from the distribution $p_{\mathbf{z}}(\mathbf{z})$. Therefore the objective becomes: $\min\limits_{\boldsymbol{\theta}} \int p_{\mathbf{z}}(\mathbf{z}) \log(1 - D_{\boldsymbol{\varphi}}(G_{\boldsymbol{\theta}}(\mathbf{z}))d\mathbf{z} = \min\limits_{\boldsymbol{\theta}} \mathbb{E}_{\mathbf{z} \sim p_{\mathbf{z}}(\mathbf{z})}[\log(1 - D_{\boldsymbol{\varphi}}(G_{\boldsymbol{\theta}}(\mathbf{z})))]$

Table of Contents

1. Motivation
2. What are GANs?
3. Generator model
4. Discriminator model
5. Generator Objective
6. Discriminator Objective
7. Combined objective function
8. Problem in gradient
9. Modified objective
10. Algorithm for training GANs
11. Theoritical Analysis
12. DeepI2I: Summary
13. References
14. Acknowledgements

Discriminator Objective

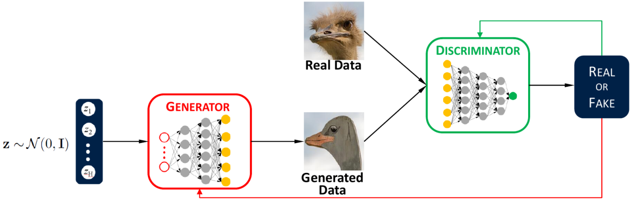


- The discriminator should assign high score to real images $\max_{\varphi} \mathbb{E}_{\mathbf{x}\sim p_{data}}[\log D_{\varphi}(\mathbf{x})]$
- Low scores (minimize) to generated images $\min_{\varphi} \mathbb{E}_{\mathbf{z}\sim p_z}[\log D_{\varphi}(G_{\boldsymbol{\theta}}(\mathbf{z}))]$ or $\max_{\varphi} \mathbb{E}_{\mathbf{z}\sim p_z}[\log(1 - D_{\varphi}(G_{\boldsymbol{\theta}}(\mathbf{z})))]$
- The combined discriminator objective function can be written as: $\max_{\varphi} \mathbb{E}_{\mathbf{x}\sim p_{data}}[\log D_{\varphi}(\mathbf{x})] + \mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}}}[\log(1 - D_{\varphi}(G_{\boldsymbol{\theta}}(\mathbf{z})))]$

Table of Contents

1. Motivation
2. What are GANs?
3. Generator model
4. Discriminator model
5. Generator Objective
6. Discriminator Objective
7. Combined objective function
8. Problem in gradient
9. Modified objective
10. Algorithm for training GANs
11. Theoritical Analysis
12. DeepI2I: Summary
13. References
14. Acknowledgements

Combined objective function

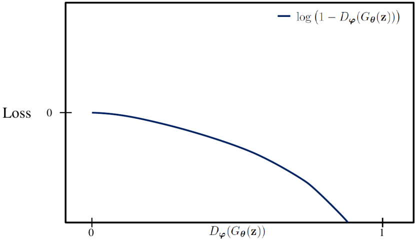


- The generator and discriminator's objective combined yeilds a minmax problem: $\min_{\boldsymbol{\theta}} \max_{\boldsymbol{\varphi}} \mathbb{E}_{\mathbf{x} \sim p_{data}}[\log \ D_{\boldsymbol{\varphi}}(\mathbf{x})] + \mathbb{E}_{\mathbf{z} \sim p_{\mathbf{z}}}[\log(1 - D_{\boldsymbol{\varphi}}(G_{\boldsymbol{\theta}}(\mathbf{z})))]$
- The first expectation is independent of $\boldsymbol{\theta}$.
- The second expectation is minimized w.r.t. $\boldsymbol{\theta}$ and maximized w.r.t $\boldsymbol{\varphi}$, hence, the generator wants to minimize the second term while the discriminator wants to maximize it.

Table of Contents

1 Motivation
2 What are GANs?
3 Generator model
4 Discriminator model
5 Generator Objective
6 Discriminator Objective
7 Combined objective function
8 Problem in gradient
9 Modified objective
10 Algorithm for training GANs
11 Theoritical Analysis
12 DeepI2I: Summary
13 References
14 Acknowledgements

Problem in gradient

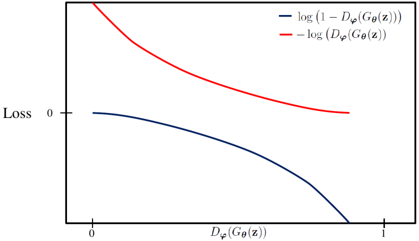


- Early on in training it is more likely that $D_\varphi(G_\theta(\mathbf{z})) \approx 0$ as the generator has not learnt much and it is easy for the discriminator to identify the difference between real and fake samples.
- In such a situation, the gradient of $\log(1 - D_\varphi(G_\theta(\mathbf{z})))$ is close to zero. The generator does not learn much in the beginning and there is little change in $\boldsymbol{\theta}$.

Table of Contents

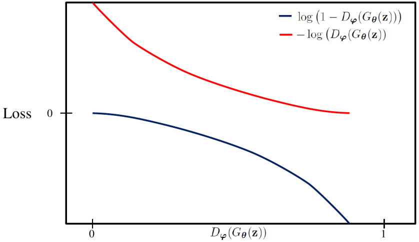
1. Motivation
2. What are GANs?
3. Generator model
4. Discriminator model
5. Generator Objective
6. Discriminator Objective
7. Combined objective function
8. Problem in gradient
9. **Modified objective**
10. Algorithm for training GANs
11. Theoritical Analysis
12. DeepI2I: Summary
13. References
14. Acknowledgements

Modified objective



- Originally we were performing minimization w.r.t. the generator parameters $\boldsymbol{\theta}$: $\min_{\boldsymbol{\theta}} \mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}}(\mathbf{z})}[\log(1 - D_{\varphi}(G_{\boldsymbol{\theta}}(\mathbf{z})))]$
- Now, we use a modified version of the above (minimization) objective: $\min_{\boldsymbol{\theta}} \mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}}(\mathbf{z})}[-\log(D_{\varphi}(G_{\boldsymbol{\theta}}(\mathbf{z})))] \equiv \max_{\boldsymbol{\theta}} \mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}}(\mathbf{z})}[\log(D_{\varphi}(G_{\boldsymbol{\theta}}(\mathbf{z})))]$

Modified objective



- We modify the objective function to $-\log(D_\varphi(G_\theta(\mathbf{z})))$
- This has large gradient when $D_\varphi(G_\theta(\mathbf{z})) \approx 0$
- This also enables the generator to learn more in the early training period.

Table of Contents

1. Motivation
2. What are GANs?
3. Generator model
4. Discriminator model
5. Generator Objective
6. Discriminator Objective
7. Combined objective function
8. Problem in gradient
9. Modified objective
10. Algorithm for training GANs
11. Theoritical Analysis
12. DeepI2I: Summary
13. References
14. Acknowledgements

Algorithm for training GANs

Algorithm 1 Minibatch stochastic gradient descent training of generative adversarial nets. The number of steps to apply to the discriminator, k, is a hyperparameter. We used $k = 1$, the least expensive option, in our experiments.

for number of training iterations **do**

for k steps **do**

- Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \dots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
- Sample minibatch of m examples $\{\boldsymbol{x}^{(1)}, \dots, \boldsymbol{x}^{(m)}\}$ from data generating distribution $p_{\text{data}}(\boldsymbol{x})$.
- Update the discriminator by ascending its stochastic gradient:

$$\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^{m} \left[\log D\left(\boldsymbol{x}^{(i)}\right) + \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right) \right].$$

end for

- Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \dots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
- Update the generator by descending its stochastic gradient:

$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^{m} \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right).$$

end for

The gradient-based updates can use any standard gradient-based learning rule. We used momentum in our experiments.

Table of Contents

1. Motivation
2. What are GANs?
3. Generator model
4. Discriminator model
5. Generator Objective
6. Discriminator Objective
7. Combined objective function
8. Problem in gradient
9. Modified objective
10. Algorithm for training GANs
11. Theoritical Analysis
12. DeepI2I: Summary
13. References
14. Acknowledgements

Theoritical Analysis

- Objective: $\min_{\mathbf{G}} \max_{\mathbf{D}} \mathbb{E}_{\mathbf{x}\sim p_{data}(\mathbf{x})}[\log\ D_{\varphi}(\mathbf{x})] + \mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}}}(\mathbf{z})[\log(1 - D_{\varphi}(G_{\boldsymbol{\theta}}(\mathbf{z})))]$
- We can re-write the objective function as:
  $\min_{\mathbf{G}} \max_{\mathbf{D}}\ \int_{\mathbf{x}} p_{data}(\mathbf{x}) \log\ D_{\varphi}(\mathbf{x})d\mathbf{x} + \int_{\mathbf{z}} p_{\mathbf{z}}(\mathbf{z}) \log(1 - D_{\varphi}(G_{\boldsymbol{\theta}}(\mathbf{z})))d\mathbf{z}$
  $\min_{\mathbf{G}} \max_{\mathbf{D}}\ \int_{\mathbf{x}} p_{data}(\mathbf{x}) \log\ D_{\varphi}(\mathbf{x})d\mathbf{x} + \int_{\mathbf{x}} p_{\mathbf{G}}(\mathbf{x}) \log(1 - D_{\varphi}(\mathbf{x}))d\mathbf{x}$
- Revised objective can be written as: $\min_{\mathbf{G}} \max_{\mathbf{D}} \mathcal{M}(G_{\boldsymbol{\theta}}, D_{\varphi})$
- Where: $\mathcal{M}(G_{\boldsymbol{\theta}}, D_{\varphi}) = \int_{\mathbf{x}}(p_{data}(\mathbf{x}) \log\ D_{\varphi}(\mathbf{x}) + p_{\mathbf{G}}(\mathbf{x}) \log(1 - D_{\varphi}(\mathbf{x})))d\mathbf{x}$
- For a given generator $G_{\boldsymbol{\theta}}$, we need the discriminator D_{φ} which maximizes the objective.
- The objective is maximized when the integrand is maximized. Differentitating the integrand w.r.t. D_{φ} yields:
  $$\frac{d}{d(D_{\varphi}(\mathbf{x}))}(p_{data}(\mathbf{x})\ \log D_{\varphi}(\mathbf{x}) + p_{\mathbf{G}}(\mathbf{x}) \log(1 - D_{\varphi}(\mathbf{x}))) = 0$$

Theoritical Analysis

$$(p_{data}(\mathbf{x})\frac{1}{D_{\varphi}(\mathbf{x})} - p_{\mathbf{G}}(\mathbf{x})\frac{1}{1-D_{\varphi}(\mathbf{x})})\frac{d}{d(D_{\varphi}(\mathbf{x}))}D_{\varphi}(\mathbf{x}) = 0$$

$$p_{data}(\mathbf{x})\frac{1}{D_{\varphi}(\mathbf{x})} = p_{\mathbf{G}}(\mathbf{x})\frac{1}{1-D_{\varphi}(\mathbf{x})}$$

$$D_{\varphi}(\mathbf{x}) = \frac{p_{data}(\mathbf{x})}{p_{\mathbf{G}}(\mathbf{x}) + p_{data}(\mathbf{x})}$$

- Therefore for optimal discriminator we have: $D^*{}_{\varphi}(\mathbf{x}) = \dfrac{p_{data}(\mathbf{x})}{p_{\mathbf{G}}(\mathbf{x}) + p_{data}(\mathbf{x})}$
- Let $\mathcal{C}(\mathbf{G}_{\boldsymbol{\theta}}) = \max \mathcal{M}(\mathbf{G}_{\boldsymbol{\theta}}, \mathbf{D}_{\varphi}) = \mathcal{M}(\mathbf{G}_{\boldsymbol{\theta}}, \mathbf{D}^*{}_{\varphi})$. Therefore, we have:
  $\mathcal{C}(\mathbf{G}_{\boldsymbol{\theta}}) =$
  $$\int \left(p_{data}(\mathbf{x}) \log \left(\frac{p_{data}(\mathbf{x})}{p_{\mathbf{G}}(\mathbf{x}) + p_{data}(\mathbf{x})} \right) + p_G(\mathbf{x}) \log \left(1 - \frac{p_{data}(\mathbf{x})}{p_{\mathbf{G}}(\mathbf{x}) + p_{data}(\mathbf{x})} \right) \right) d\mathbf{x}.$$
  $$= \int \left(p_{data}(\mathbf{x}) \log \left(\frac{p_{data}(\mathbf{x})}{p_{\mathbf{G}}(\mathbf{x}) + p_{data}(\mathbf{x})} \right) + p_G(\mathbf{x}) \log \left(\frac{p_{\mathbf{G}}(\mathbf{x})}{p_{\mathbf{G}}(\mathbf{x}) + p_{data}(\mathbf{x})} \right) \right) d\mathbf{x}$$

Theoritical Analysis

$$= \int (p_{data}(\mathbf{x}) \left(\log 2 + \log \left(\frac{p_{data}(\mathbf{x})}{p_{\mathbf{G}}(\mathbf{x}) + p_{data}(\mathbf{x})}\right)\right) +$$

$$p_G(\mathbf{x}) \left(\log 2 + \log \left(\frac{p_{\mathbf{G}}(\mathbf{x})}{p_{\mathbf{G}}(\mathbf{x}) + p_{data}(\mathbf{x})}\right)\right) - (p_{data}(\mathbf{x}) + p_{\mathbf{G}}(\mathbf{x})) \log 2) d\mathbf{x}$$

$$= \int \left(p_{data}(\mathbf{x}) \left(\log \frac{p_{data}(\mathbf{x})}{\frac{p_{\mathbf{G}}(\mathbf{x}) + p_{data}(\mathbf{x})}{2}} \right) + p_G(\mathbf{x}) \left(\log \frac{p_{\mathbf{G}}(\mathbf{x})}{\frac{p_{\mathbf{G}}(\mathbf{x}) + p_{data}(\mathbf{x})}{2}} \right) \right) d\mathbf{x} -$$

$$\log 2 \int (p_{data}(\mathbf{x}) + p_{\mathbf{G}}(\mathbf{x}))\ d\mathbf{x}$$

$$= KL\left(p_{data}(\mathbf{x}) || \frac{p_{\mathbf{G}}(\mathbf{x}) + p_{data}(\mathbf{x})}{2}\right) + KL\left(p_{\mathbf{G}}(\mathbf{x}) || \frac{p_{\mathbf{G}}(\mathbf{x}) + p_{data}(\mathbf{x})}{2}\right) - 2\ \log\ 2$$

Theorem

The global minimum of $\mathcal{C}(\mathbf{G}_{\boldsymbol{\theta}})$ is achieved if and only if $p_{\mathbf{G}} = p_{data}$.

Theoritical Analysis[2]

$$\mathcal{C}(\mathbf{G}_{\boldsymbol{\theta}}) =$$

$$KL\left(p_{data}(\mathbf{x})||\frac{p_{\mathbf{G}}(\mathbf{x}) + p_{data}(\mathbf{x})}{2}\right) + KL\left(p_{\mathbf{G}}(\mathbf{x})||\frac{p_{\mathbf{G}}(\mathbf{x}) + p_{data}(\mathbf{x})}{2}\right) - 2\log 2$$

- If $p_{\mathbf{G}} = p_{data}$, then the minimum of $\mathcal{C}(\mathbf{G}_{\boldsymbol{\theta}})$ is attained
- For $p_{\mathbf{G}} = p_{data}$, we have $\min_G \mathcal{C}(\mathbf{G}_{\boldsymbol{\theta}}) = -\log 4$ as $KL(p_G||p_G) = 0$.
- For $p_{\mathbf{G}} \neq p_{data}$, we have $\min_G \mathcal{C}(\mathbf{G}_{\boldsymbol{\theta}}) \geq -\log 4$ as $KL(.||.) \geq 0$.
- If the minimum of $\mathcal{C}(\mathbf{G}_{\boldsymbol{\theta}})$ is achieved, then $p_{\mathbf{G}} = p_{data}$

  $$\mathcal{C}(\mathbf{G}_{\boldsymbol{\theta}}) =$$

  $$KL\left(p_{data}(\mathbf{x})||\frac{p_{\mathbf{G}}(\mathbf{x}) + p_{data}(\mathbf{x})}{2}\right) + KL\left(p_{\mathbf{G}}(\mathbf{x})||\frac{p_{\mathbf{G}}(\mathbf{x}) + p_{data}(\mathbf{x})}{2}\right) - \log 4$$

  $$= 2JS(p_{data}(\mathbf{x})||p_{\mathbf{G}}(\mathbf{x})) - \log 4$$
- $2JS(p_{data}(\mathbf{x})||p_{\mathbf{G}}(\mathbf{x})) = 0$ only when $p_{\mathbf{G}} = p_{data}$

[2] From the notes of Dripta Mj, RKMVERI

Table of Contents

1 Motivation
2 What are GANs?
3 Generator model
4 Discriminator model
5 Generator Objective
6 Discriminator Objective
7 Combined objective function
8 Problem in gradient
9 Modified objective
10 Algorithm for training GANs
11 Theoritical Analysis
12 DeepI2I: Summary
13 References
14 Acknowledgements

DeepI2I: Summary

- *Image to Image (I2I)* translation is used in the field of Computer Graphics (CG), specially in the movie industries.
- Traditionally this is a labour intensive process, and the proposed technique can be applied to automatic translation of faces/objects.
- I2I translations shows inferior performance when translations between classes requires large shape changes.
- Learn the model by leveraging hierarchical features:
  - Structural information contained in the shallow layers.
  - Semantic information extracted from the deep layers.
- Implemented a novel transfer learning method by transferring knowledge from pre-trained GANs, enabling learning on small datasets.

DeepI2I: Summary

- Leverage the discriminator of pretrained GAN to initialize the encoder and discriminator, and leverage the pretrained generator to initialize the generator of their model.
- Introduced Adaptor network to address the alignment problem between encoder and decoder when using knowledge transfer.
- They are first to do I2I trainslation over 1000 classes in animal faces, birds and food datasets.
- They qualitatively and quantitatively showed that transfer learning significantly improves the performance of I2I systems for small datasets.

DeepI2I: Model

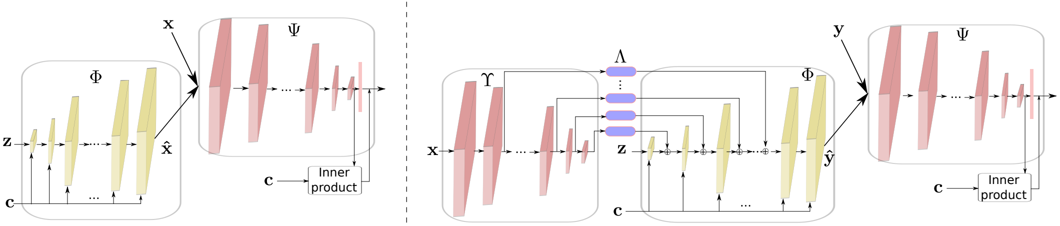


Figure 3: *Left*: the traditional form of conditional GAN (i.e., BigGAN) which contains the generator Φ and the discriminator Ψ. *Right*: the proposed DeepI2I method based on conditional GAN (left). The method consists of four terms: the encoder Υ, the adaptor Λ, the generator Φ and the discriminator Ψ. The encoder Υ is initialized by pre-trained discriminator (left), as well as both the generator Φ and the discriminator Ψ by pre-trained GANs (left). The adaptor Λ aims to align the pre-trained encoder Υ and the pre-trained generator Ψ.

DeepI2I: Results

Qualitative comparison on animal faces and foods. The input images are in the first column and the remaining columns show the class-specific translated images.

Qualitative results of DeepI2I. The input image is in the first column and the remaining column show the class-specific outputs. For each specific target class, we show two images.

Unseen I2I translation: the input image is mapped into eight animal faces with StarGAN and DeepI2I.

Table of Contents

1. Motivation
2. What are GANs?
3. Generator model
4. Discriminator model
5. Generator Objective
6. Discriminator Objective
7. Combined objective function
8. Problem in gradient
9. Modified objective
10. Algorithm for training GANs
11. Theoritical Analysis
12. DeepI2I: Summary
13. References
14. Acknowledgements

References I

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. Generative adversarial nets. In Zoubin Ghahramani, Max Welling, Corinna Cortes, Neil D. Lawrence, and Kilian Q. Weinberger, editors, Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Mon- treal, Quebec, Canada, pages 2672–2680, 2014. URL: `https://proceedings.neurips.cc/paper/2014/hash/5ca3e9b122f61f8f06494c97b1afccf3-Abstract.html`.

Yunjey Choi, Min-Je Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018, pages 8789–8797. Computer Vision Foundation / IEEE Computer Society, 2018. doi: 10.1109/CVPR.2018.00916. URL: `http://openaccess.thecvf.com/content_cvpr_2018/html/Choi_StarGAN_Unified_Generative_CVPR_2018_paper.html`.

Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal unsupervised image-to-image translation. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, Computer Vision – ECCV 2018, pages 179–196, Cham, 2018. Springer International Publishing. ISBN 978-3-030-01219-9.

Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. In 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019.OpenReview.net, 2019. URL: `https://openreview.net/forum?id=B1xsqj09Fm`

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 4396–4405, 2019.doi: 10.1109/CVPR.2019.00453.

References II

Hsin-Ying Lee, Hung-Yu Tseng, Qi Mao, Jia-Bin Huang, Yu-Ding Lu, Maneesh Singh, and Ming-Hsuan Yang. DRIT++: diverse image-to-image translation via disentangled representations. Int. J. Comput. Vis., 128(10): 2402–2417, 2020. doi: 10.1007/s11263-019-01284-z. URL: https://doi.org/10.1007/s11263-019-01284-z.

Justin N. M. Pinkney and Doron Adler. Resolution dependent GAN interpolation for controllable image synthesis between domains. CoRR, abs/2010.05334, 2020. URL: https://arxiv.org/abs/2010.05334.

Yaxing Wang, Abel Gonzalez-Garcia, Joost van de Weijer, and Luis Herranz. SDIT: scalable and diverse cross-domain image translation. In Laurent Amsaleg, Benoit Huet, Martha A. Larson, Guillaume Gravier, Hayley Hung, Chong-Wah Ngo, and Wei Tsang Ooi, editors, Proceedings of the 27th ACM International Conference on Multimedia, MM 2019, Nice, France, October 21-25, 2019, pages 1267–1276. ACM, 2019. doi: 10.1145/3343031.3351004. URL: https://doi.org/10.1145/3343031.3351004.

Yaxing Wang, Abel Gonzalez-Garcia, David Berga, Luis Herranz, Fahad Shahbaz Khan, and Joost van de Weijer. Minegan: Effective knowledge transfer from gans to target domains with few images. In 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020, pages 9329–9338. Computer Vision Foundation / IEEE, 2020. doi: 10.1109/CVPR42600.2020.00935. URL: https://openaccess.thecvf.com/content_CVPR_2020/html/Wang_MineGAN_Effective_Knowledge_Transfer_From_GANs_to_Target_Domains_With_CVPR_2020_paper.html.

Xiaoming Yu, Yuanqi Chen, Shan Liu, Thomas H. Li, and Ge Li. Multi-mapping image-to-image translation via learning disentanglement. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché- Buc, Emily B. Fox, and Roman Garnett, editors, Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada, pages 2990–2999, 2019. URL: https://proceedings.neurips.cc/paper/2019/hash/5a142a55461d5fef016acfb927fee0bd-Abstract.html.

Table of Contents

1. Motivation
2. What are GANs?
3. Generator model
4. Discriminator model
5. Generator Objective
6. Discriminator Objective
7. Combined objective function
8. Problem in gradient
9. Modified objective
10. Algorithm for training GANs
11. Theoritical Analysis
12. DeepI2I: Summary
13. References
14. Acknowledgements

Acknowledgements

It is ritual that scholars express their gratitude to their instructors. This acknowledgement is very special to me to express my deepest sense of gratitude and pay respect to my instructor, **P. Balamurugan**, Department of Industrial Engineering and Operations Research, for his constant encouragement, guidance, supervision, and support throughout the completion of my project. His close scrutiny, constructive criticism, and intellectual insight have immensely helped me in every stage of my work. I would like to thank him for patiently answering my often-naive questions related to Computer Vision.

I would like to thank my thesis supervisor **Suyash P. Awate** for arranging free compute resources and help me suggest such a wonderful course. I would also like to thank the awesome T.A.s, **Vaibhav Singh Panwar** and **Akash Saha** for the stimulating discussions that they had shared with me during the Deep Learning Theory and Practices course. Finally I would like to thank my father **Dr. Jadab Kumar Pal**, Indian Statistical Institute, for supporting me.

Any Questions . . . ?

Thank You

22d1594@iitb.ac.in
jimutbahanpal@yahoo.com