

# PAPER ANALYSIS

Presented by Yannis He



Paper: **Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks**

Conference: ICCV 2017

Authors:

Jun-Yan Zhu, Taesung Park, Phillip Isola, Alexei A. Efros | Berkeley AI Research (BAIR)  
laboratory, UC Berkeley

<https://arxiv.org/pdf/1703.10593.pdf>

Abstract:

- Contribution:
  - Approach for learning to translate an image from a source domain to a target domain in the absence of paired examples.
    - Learn mapping,  $G: X \rightarrow Y$ , such that distribution of image from  $G(X)$  is indistinguishable from distribution  $Y$  using an adversarial loss.
  - Found that this *highly under-constrained* mapping can be inverse:
    - There exists  $F: Y \rightarrow X$ , such that  $F(G(X)) \approx X$

Intro:

- The translation can be made in the absence of any paired training examples
  - Assume there are underlying relationship between domains
  - Lack of supervision in forms of pairs
    - But can exploit supervision at level of sets
- Motivation:
  - Only a couple of datasets exist for semantic segmentation task
    - And those dataset are small
  - Obtaining input-output pairs are expensive
    - Some output are even not well-defined
      - E.g. zebra  $\leftrightarrow$  horse

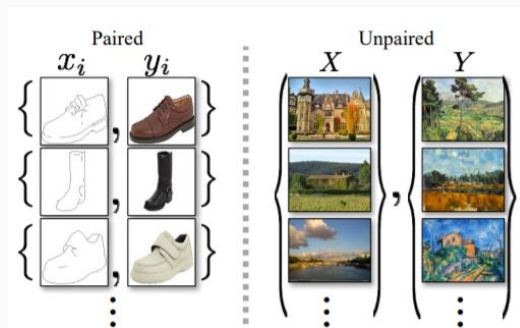


Figure 2: *Paired* training data (left) consists of training examples  $\{x_i, y_i\}_{i=1}^N$ , where the correspondence between  $x_i$  and  $y_i$  exists [22]. We instead consider *unpaired* training data (right), consisting of a source set  $\{x_i\}_{i=1}^N$  ( $x_i \in X$ ) and a target set  $\{y_j\}_{j=1}^M$  ( $y_j \in Y$ ), with no information provided as to which  $x_i$  matches which  $y_j$ .

- Ideas:
  - First attempts
    - Using adversary train, to train a mapping  $G: X \rightarrow Y$ , where  $\tilde{y} = G(x)$  &  $x \in X$ , such that  $\tilde{y}$  is indistinguishable from  $y \in Y$ 
      - Ideally, we just obtained an  $\tilde{Y}$  that distributes identically to  $Y$
      - However, such translation does not guarantee a meaningful pair-up
        - It is difficult to optimize adversarial objective in isolation
          - Often lead to “mode collapse”
    - Second attempts (our proposal)
      - Using two inverted translator:  $G: X \rightarrow Y$  &  $F: Y \rightarrow X$ , which should lead the mapping to be bijection
        - Train both mapping simultaneously
        - with a cycle consistency loss along with a adversarial loss on domains  $X$  and  $Y$

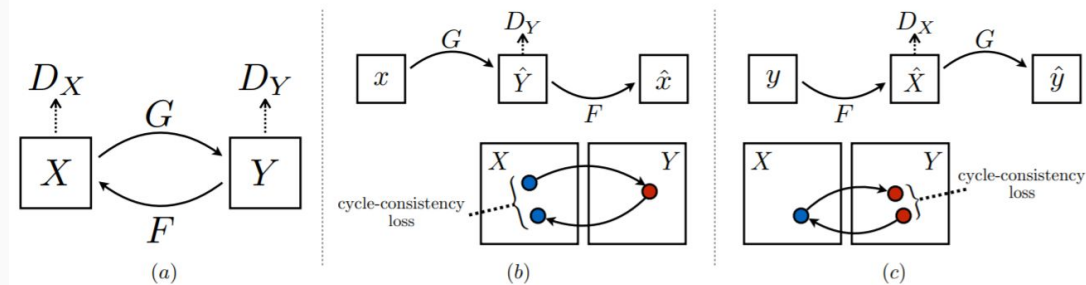


Figure 3: (a) Our model contains two mapping functions  $G : X \rightarrow Y$  and  $F : Y \rightarrow X$ , and associated adversarial discriminators  $D_Y$  and  $D_X$ .  $D_Y$  encourages  $G$  to translate  $X$  into outputs indistinguishable from domain  $Y$ , and vice versa for  $D_X$  and  $F$ . To further regularize the mappings, we introduce two *cycle consistency losses* that capture the intuition that if we translate from one domain to the other and back again we should arrive at where we started: (b) forward cycle-consistency loss:  $x \rightarrow G(x) \rightarrow F(G(x)) \approx x$ , and (c) backward cycle-consistency loss:  $y \rightarrow F(y) \rightarrow G(F(y)) \approx y$

- Generative Adversarial Networks (GANs)
- Image-to-Image Translation
- Unpaired Image-to-Image Translation (some other approaches)
  - Bayesian framework including a prior based on a patch-based Markov random field
  - Weight-sharing strategy to learn common representation across domains
    - CoGAN
    - Cross-modal scene networks
  - Variational Autoencoders (VAEs) + GANs
  - \* different from above approach, the proposed method does not rely on any task-specific, predefined similarity function, nor we assumed both domain lie in the same low-dimensional embedding space.
    - I.e. the proposed approach is **general-purpose solution**
- Cycle Consistency
  - Using transitivity as a way to regularize structured data
  - Back translation and reconciliation
    - Used in language translation by human and machines
  - High-order cycle consistency
    - Used from motion, 3D shape matching, co-segmentation, dense semantic alignment, etc.
  - Cycle consistency loss: a way of using transitivity to supervise CNN training (most similar to proposed method)
- Neural Style Transfer
  - Learning mapping between two collections rather than two specific images

- Setup:

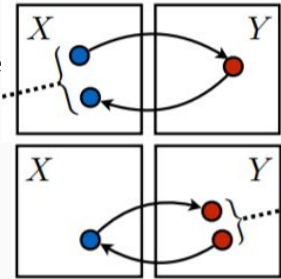
- Two inverted mappings:  $G: X \rightarrow Y$  &  $F: Y \rightarrow X$
- Two adversarial discriminators:  $D_x$  and  $D_y$ 
  - $D_x$  aims to distinguish between images  $\{x\}$  and translated images  $\{F(y)\}$
  - $D_y$  aims to distinguish between images  $\{y\}$  and translated images  $\{G(x)\}$
- Objectives:
  - Adversarial losses: matching the distribution of generated images to the data distribution in target domains

$$\begin{aligned} \mathcal{L}_{\text{GAN}}(G, D_Y, X, Y) = & \mathbb{E}_{y \sim p_{\text{data}}(y)} [\log D_Y(y)] \\ & + \mathbb{E}_{x \sim p_{\text{data}}(x)} [\log(1 - D_Y(G(x))], \end{aligned} \quad (1)$$

- Cycle consistency losses: prevent the learned mapping from contradicting each other
  - Requires the functions to be cycle-consistent

$$\begin{aligned} \mathcal{L}_{\text{cyc}}(G, F) = & \mathbb{E}_{x \sim p_{\text{data}}(x)} [\|F(G(x)) - x\|_1] \\ & + \mathbb{E}_{y \sim p_{\text{data}}(y)} [\|G(F(y)) - y\|_1]. \end{aligned} \quad (2)$$

Forward cycle consistency



Backward cycle consistency

- Overall objectives:

$$\begin{aligned} \mathcal{L}(G, F, D_X, D_Y) = & \mathcal{L}_{\text{GAN}}(G, D_Y, X, Y) \\ & + \mathcal{L}_{\text{GAN}}(F, D_X, Y, X) \\ & + \lambda \mathcal{L}_{\text{cyc}}(G, F), \end{aligned} \quad (3)$$

$$G^*, F^* = \arg \min_{G, F} \max_{D_x, D_y} \mathcal{L}(G, F, D_X, D_Y). \quad (4)$$

- Network Architecture:
  - 3 Convolutions, several residual blocks
  - 2 fractionally-strided convolution with  $\frac{1}{2}$  stride
  - 1 convolution that maps features to RGB
- Implementation:
  - Use 6 blocks for 128 x 128 images and 9 blocks for 256 x 256 and higher-resolution training images
  - Use instance normalization
  - Use 70 x 70 PatchGANs for discriminator networks
    - To classify whether 70 x 70 overlapping image patches are real or fake
- Training
  - Two training techniques to stabilize our model training procedure
    - For  $L_{\text{GAN}}$  (eqn 1), we use least-square instead of negative log likelihood since the former is more stable in this case
    - To reduce model oscillation, we update discriminator using a history of generated images rather than the one produced by the latest generators. (the authors keep an image buffer that stores 50 previously created images)
  - $\lambda = 10$  is used in eqn 3
  - Adam solver
  - batch\_size = 1
  - lr = 0.0002 (for first 100 epochs) and linearly decay to 0 over the next 100 epochs

- Evaluation:
  - Metrics: pix2pix, FCN Score, Semantic segmentation metrics
  - Baseline: CoGAN, SimGAN, Feature loss + GAN, BiGAN/ALI, pix2pix
- Analysis:
  - Both GAN loss and cycle-consistency loss are important
  - Bidirectional are important. Single directional leads to mode collapse
  - Image resolutions would not drop during the transformation
- Applications:
  - Collection style transfer
  - Object transfiguration
  - Season transfer
  - Photo generation from paintings
  - Photo enhancement
- Limitations and Discussion:
  - Results are far from uniformly positive
    - Good at: tasks involving color and texture changes
    - Not good at: tasks involving geometric changes
  - Some failure caused by distribution characteristics of the training datasets
  - Lingering gap between paired training data vs unpaired method
    - Potential solution: integrating weak or semi-supervised data