Reducing Steganography In Cycle-consistency GANs

Horia Porav
Oxford Robotics Institute
University of Oxford
horia@robots.ox.ac.uk

Valentina Musat
Oxford Brookes University
Oxford
valentina.musat-2016@brookes.ac.uk

Paul Newman
Oxford Robotics Institute
Oxford
pnewman@robots.ox.ac.uk

Abstract

In this work we present a simple method of improving the suitability of data generated using cycle-consistency GANs in the context of day-to-night domain adaptation. While CycleGANs produce visually pleasing outputs, they also encode hidden (steganographic) information about the source domain in the generated images, which makes them less suitable as training data generators. We reduce the amount of steganographic information hidden in the generated images by introducing an end-to-end differentiable image denoiser in between the two generators. The role of the denoiser is to strip away the high frequency, low amplitude encoded information, making it harder for the generators to hide information that is invisible to the discriminator. We benchmark the suitability of data generated using our simple method in the context of simple domain adaptation for semantic segmentation, comparing with standard CycleGAN, MUNIT and DRIT and show that our method yields better performance.

1. Introduction

Cycle-consistency GANs have become popular ever since their introduction, and have been applied for artistic purposes and domain adaptation in fields like robotics and medical image analysis, where acquiring paired images from other domains is at best expensive, and at worst impossible. It was recently shown [3], however, that generated images contain hidden (steganographic) information about the source domain, and that the generators can end up "hallucinating" content in order to fool the discriminators. The abject simplicity that has made CycleGANs so popular is also its weak point: in order to minimize the reconstruction loss, the two generators cooperatively develop a common, high-frequency encoding for source domain information which is often not picked up by the discriminators. This inclusion of encoded source data raises concerns regarding the suitability of the generated data for domain adaptation training or fine-tuning, and in critical setups such as medical analysis or autonomous driving. A simple example is that of generating night time images from day time images, for training a semantic segmentation task. The segmentation model will learn to decode the steganographic daytime information, which won't be present in the real night time images, leading to a lower bound on performance, spurious results, and an inefficient use of model parameters.

Figure 1: Our method reduces the amount of information hidden in the generated images of cycle-consistency GANs. Top row, from left to right: input day time image, fake night time image and reconstructed day image generated by standard CycleGAN. Bottom row: input day time image, fake night time image and reconstructed day image generated by our improved CycleGAN. In contrast to the standard CycleGAN model, our method drastically reduces the amount of steganographic information hidden in the generated image, and as such is unable to accurately reconstruct the input image.

In this work, inspired by the study of [6] in combating adversarial noise, we present a simple method of reducing the amount of cooperation between the two generators by introducing an end-to-end differentiable image denoiser in between the two generators. The role of the denoiser is to strip away the high frequency, low amplitude information, making it harder for the generators to hide information that is invisible to the discriminator, without having to employ an over-parameterised discriminator that would

destabilize training. We benchmark the suitability of data generated using our method in the context of day-to-night domain adaptation for semantic segmentation. To focus on the improvement brought by having better data, we simply apply day-to-night style transfer to the training split of the Cityscapes [4] dataset and train a semantic segmentation model (Deeplab V3+[2]) on the generated data. We test the trained segmentation models on the test split (no train split available) of the NightTimeDriving[5] dataset. We compare our method with standard CycleGAN[12], MUNIT[8] and DRIT[9], and show that the semantic segmentation model trained on data generated using our method has better performance.

2. Related Work

2.1. Cycle-consistency GANs and steganography

CycleGANs are a popular tool for unpaired image-to-image translation, as opposed to previous methods that require image pairs from both domains. Using an adversarial loss to train a generative model implies learning a transformation $G_{AB} : A \rightarrow B$ such that the distribution of images from domain A rendered in domain B cannot be distinguished from the distribution of real images from domain B. However, this poses a problem, since there can be many images whose distribution is similar to domain B, but with meaningless content. In order to preserve the structure of the image, a cycle consistency constraint is employed, by introducing a second transformation model $G_{BA} : B \rightarrow A$ [11].

Although advantageous, CycleGANs pose a major drawback, as shown in a study by [3], where aerial photographs are translated to maps. The authors find that the model hides information about the aerial image into the generated map (even in solid-color areas), ensuring that aerial images will be reconstructed back with the finest details. By adding high-frequency noise, they compose a map b^* (that is visually similar to an original map b_0) and show that the generator G_{BA} can be forced to recreate a particular aerial photograph a^* from b^*. As the difference between the b^* and b_0 required by the generator in order to produce a^* decreases during training, they conclude that G_{BA} is colluding with G_{AB}'s adversarial attack, where G_{AB} encodes information about the source image and G_{BA} learns to decode the hidden information and recreate the source image in such detail that would otherwise be difficult to reconstruct, since there are many aerial photos that can correspond to one map.

2.2. Reducing steganography in cycle-consistency GANs

In a high-fidelity image to image translation task, [7] tackle steganography by ensuring an uncooperative setup and adding a residual/style path. In the traditional cooperative setup, the networks are trained on each other's output, which means that if the second network is able to compensate for the first network's error, then the first network doesn't need to improve. The proposed solution is to only train the networks when input is real, instead of both real and generated. Their architecture consists of two networks: a disentangler network D, which splits the image from domain V into C and R, where C is another image domain and R is the residual between the two domains, and an entangler network E that merges C and R back to V. In the first learning cycle, D is updated based on the reconstruction error of v, where $D(v) = (c', r')$ and $E(D(v)) = v'$, whereas in the second cycle, E is updated based on the reconstructions of r and c, where E generates $E(c, r) = v'$ and D disentangles $D(v') = (c', r')$. In both cycles the input data is real: in the first cycle, v is an image from domain V, in the second cycle, c is an image from domain C and r is generated from a random v.

As one characteristic of CycleGAN is the deterministic one-to-one mapping, [1] propose Augmented-CycleGAN in order to ensure many-to-many mappings across domains. By allowing the two generators to be conditional on a latent space Z, one could sample different $z \sim p(z)$ to generate multiple samples in the target domain, while holding the source domain fixed. However, the cycle-consistency reconstruction error implicitly assumes unimodality, since all images generated across samples z will have to be close to the original image, for the loss to be minimized. Their approach is thus to learn mappings between augmented spaces $A \times Z_b$ and $B \times Z_a$, where Z_a and Z_b are latent spaces that encode missing information: Z_a holds information about a that does not exist in the generated image b and vice-versa. Their loss function is amended to include two adversarial and two reconstruction losses for latent codes Z_a and Z_b. They check for steganography by introducing noise in the source domain images and evaluating the reconstruction error in the target domain and conclude that information is captured in the latent codes rather than being hidden in the generated images.

We draw our inspiration from [6], who study the effect of JPEG compression on adversarial perturbation and conclude that small perturbations can be eliminated using compression. We replace the non-differentiable JPEG compression with a learned denoising network.

3. Learning to reduce steganography

3.1. The image denoising network

As most of the hidden information inside cycle-consistency GANs is represented by high-frequency, low-amplitude information [3], we choose to use a differentiable denoiser to reduce this type of information. The intuition is that, within each cycle, the generators will be forced to encode a better structure-preserving output, using lower fre-

quency representations, while still minimizing discriminator losses. The architecture is presented in Figure 2.

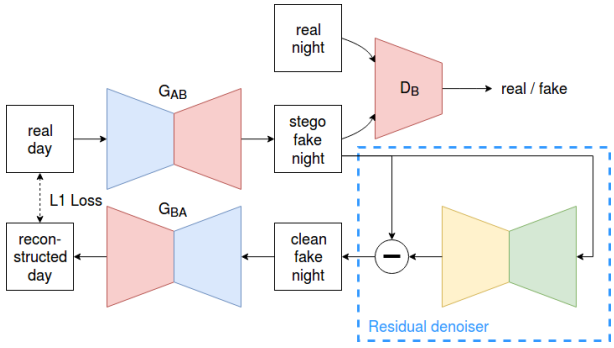


Figure 2: The architecture of one of the two cycles used to train traditional cycle-consistency GANs, augmented with a differentiable image denoising network in between the two generators. The discriminator always receives an unfiltered generated image.

We make use of a residual image denoiser R, which, invariant to the domain, takes and image I_z affected by noise z and extracts the noise component z. The original, clean image I is then reconstructed by subtracting the noise from the noise-affected image:

$$I_z = I + z \tag{1}$$

$$R(I_z) \approx z \tag{2}$$

$$I = I_z - R(I_z) \tag{3}$$

3.2. Cycle-Consistency GANs

Following [12] and using the image denoiser R described above, we use 2 generators: given an image I_A from domain A (day) and an image I_B from domain B (night), we employ generator G_AB to translate I_A to domain B and generator G_BA to translate the image back into the original domain. We add the image denoiser R in between the two generators. On the output of each generator, we apply an adversarial loss: discriminators D_B for generator G_AB, and D_A for G_BA respectively. The adversarial losses are:

$$\mathcal{L}_{\mathrm{B}_\mathrm{adv}} = (D_\mathrm{B}(G_\mathrm{AB}(I_\mathrm{A})) - 1)^2 \tag{4}$$

$$\mathcal{L}_{\mathrm{A}_\mathrm{adv}} = (D_\mathrm{A}(G_\mathrm{BA}(I_\mathrm{B})) - 1)^2 \tag{5}$$

The final adversarial objective $\mathcal{L}_\mathrm{adv}$ to be minimized becomes:

$$\mathcal{L}_\mathrm{adv} = \mathcal{L}_{\mathrm{B}_\mathrm{adv}} + \mathcal{L}_{\mathrm{A}_\mathrm{adv}} \tag{6}$$

The discriminators are trained to minimize:

$$\mathcal{L}_{\mathrm{B}_\mathrm{disc}} = (D_\mathrm{B}(I_\mathrm{B}) - 1)^2 + (D_\mathrm{B}(G_\mathrm{AB}(I_\mathrm{A})))^2 \tag{7}$$

$$\mathcal{L}_{\mathrm{A}_\mathrm{disc}} = (D_\mathrm{A}(I_\mathrm{A}) - 1)^2 + (D_\mathrm{A}(G_\mathrm{BA}(I_\mathrm{B})))^2 \tag{8}$$

The final discriminator objective $\mathcal{L}_\mathrm{disc}$ to be minimized becomes:

$$\mathcal{L}_\mathrm{disc} = \mathcal{L}_{\mathrm{B}_\mathrm{disc}} + \mathcal{L}_{\mathrm{A}_\mathrm{disc}} \tag{9}$$

Using the image denoiser R, a cycle-consistency loss [12] is computed between the input images and the reconstructed images:

$$\mathcal{L}_{\mathrm{A}_\mathrm{rec}} = \|I_\mathrm{A} - \hat{I_A}\|_1 \tag{10}$$

$$\mathcal{L}_{\mathrm{B}_\mathrm{rec}} = \|I_\mathrm{B} - \hat{I_B}\|_1 \tag{11}$$

where

$$\hat{I_A} = G_{BA}(G_{AB}(I_A) - R(G_{AB}(I_A))) \tag{12}$$

$$\hat{I_B} = G_{AB}(G_{BA}(I_B) - R(G_{BA}(I_B))) \tag{13}$$

The generator objective $\mathcal{L}_\mathrm{gen}$ becomes:

$$\mathcal{L}_\mathrm{gen} = \lambda_\mathrm{rec} * \mathcal{L}_\mathrm{rec} + \lambda_\mathrm{adv} * \mathcal{L}_\mathrm{adv} \tag{14}$$

with each λ weighing the importance of individual objectives. The generators $G_\mathrm{AB}, G_\mathrm{BA}$ that minimize the complete objective are:

$$G_\mathrm{AB}, G_\mathrm{BA} = \underset{G_\mathrm{AB}, G_\mathrm{BA}, D_\mathrm{B}, D_\mathrm{A}}{\arg\min} \mathcal{L}_\mathrm{gen} + \mathcal{L}_\mathrm{disc} \tag{15}$$

4. Experimental Setup

We compare a baseline semantic segmentation model (Deeplab V3+ with Mobilenet backbone [2]) trained on the Cityscapes train split, and the same model fine-tuned with data generated using standard CycleGAN, MUNIT [8], DRIT[9] and our method. We chose these methods for their relative popularity and similar complexity to our method.

4.1. The denoising network

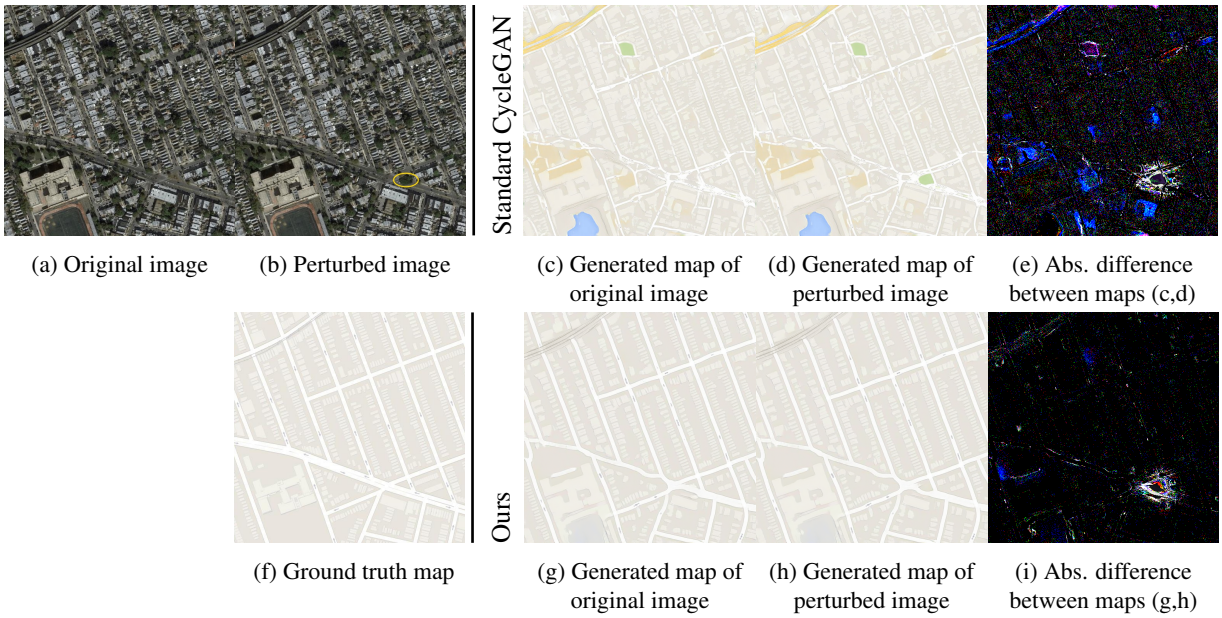
We make use of the reference implementation of DNCNN[1] [10], which is trained for blind Gaussian denoising with a large range of noise levels ($\sigma \in [0, 55]$). The network is **frozen** during our training and inference runs.

4.2. Training, generating data and fine-tuning

For training the data generation pipeline, we divide the training split of Cityscapes dataset in two equal shuffled sets of 1488 images each, and use the first set as the day-domain dataset, while using the testing split of NightTimeDriving as the night-domain dataset. For standard CycleGAN and our method, we follow the regimen of [12] and train for 50 epochs. For MUNIT, we follow the regimen described in [8] and train for 100000 iterations. Similarly, for DRIT, we follow the authors of [9] and train for 105000 iterations.

To test the quality of the generated data, we apply day-to-night style transfer to the second set of 1488 Cityscapes images, using generators trained with each of the frameworks mentioned above, producing images at both 512x256

[1] https://github.com/cszn/DnCNN



(a) Original image (b) Perturbed image (c) Generated map of original image (d) Generated map of perturbed image (e) Abs. difference between maps (c,d)

(f) Ground truth map (g) Generated map of original image (h) Generated map of perturbed image (i) Abs. difference between maps (g,h)

Figure 3: We demonstrate qualitatively that out method significantly reduces the amount and distribution of steganographic information. We add a perturbation (circled in yellow) to an input image and observe the effect it produces on the output of the generator. The first row shows results using Standard CycleGAN, while the second row shows results from our method. In contrast to Standard CycleGAN, where the perturbation is encoded **throughout** the image and with large amplitudes, our method encodes this information mostly locally, around the spatial location of the perturbation and along edges with large contrast. Note that the difference is computed over images that have **NOT** been denoised by the denoiser, and represent the raw output of the generator. Additionally, note that our results are much **closer** to the ground truth map.

and 1024x512 resolutions, and use this data to fine-tune the semantic segmentation model. Note that, for our method, the denoising network is not used at runtime. Finally, we evaluate the baseline segmentation model and each fine-tuned model on the test split of the NightTimeDriving [5] dataset, measuring Mean Intersection Over Union (mIOU). As night-time datasets with groundtruth annotations are scarce, the NightTimeDriving test split is used both as a night-time style for data generation, and as the test set for semantic segmentation. However, pollution of the generated data from the style information is minimal, and all methods are benchmarked with exactly the same setup.

5. Results

5.1. Quantitative results

While all methods performed far better than the baseline, the results presented in Table 1 indicate that our simple denoising strategy produces superior training data compared to MUNIT, DRIT and standard CycleGAN, at both resolutions. Note that our aim is to compare methods in relative terms and not to obtain absolute state-of-the-art results.

5.2. Qualitative results

In Figure 3, we show that our method significantly reduces the amount and distribution of steganographic information. Following the experiment proposed in [3], we first train our denoising-CycleGAN on an aerial-photo↔map dataset. We add a perturbation (circled in yellow) to an input image and observe the effect it produces on the output of the generator by computing the absolute difference between the map obtained from the unperturbed image and the map obtained from the perturbed image. In contrast to Standard CycleGAN, where the perturbation is encoded **throughout** the image and with large amplitudes, images generated using our method encode this information mostly locally, around the spatial location of the perturbation and along edges with large contrast.

Table 1: Semantic segmentation mIOU for all methods

	Trained on				
Training resolution	Source Domain	MUNIT	CycleGAN	DRIT	Ours
512x256	0.1305	0.1820	0.2231	0.2361	**0.2630**
1024x512	0.1305	0.2323	0.2700	0.2709	**0.2971**

6. Conclusions

We have presented a simple way to reduce steganographic information within cycle-consistency GANs and better align generated images with a target distribution. We have demonstrated this quantitatively by showing that using the improved generated images to fine-tune a semantic segmentation model leads to better performance on a real-world difficult dataset, and qualitatively by showing that the extent and amplitude of hidden information is reduced.

References

[1] Amjad Almahairi, Sai Rajeswar, Alessandro Sordoni, Philip Bachman, and Aaron C. Courville. Augmented cyclegan: Learning many-to-many mappings from unpaired data. *CoRR*, abs/1802.10151, 2018. 2

[2] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *ECCV*, 2018. 2, 3

[3] Casey Chu, Andrey Zhmoginov, and Mark Sandler. Cyclegan, a master of steganography. *CoRR*, abs/1712.02950, 2017. 1, 2, 4

[4] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3213–3223, 2016. 2

[5] Dengxin Dai and Luc Van Gool. Dark model adaptation: Semantic image segmentation from daytime to nighttime. In *IEEE International Conference on Intelligent Transportation Systems*, 2018. 2, 4

[6] Gintare Karolina Dziugaite, Zoubin Ghahramani, and Daniel M. Roy. A study of the effect of JPG compression on adversarial images. *CoRR*, abs/1608.00853, 2016. 1, 2

[7] Adam W. Harley, Shih-En Wei, Jason M. Saragih, and Katerina Fragkiadaki. Image disentanglement and uncooperative re-entanglement for high-fidelity image-to-image translation. *CoRR*, abs/1901.03628, 2019. 2

[8] Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal unsupervised image-to-image translation. In *ECCV*, 2018. 2, 3

[9] Hsin-Ying Lee, Hung-Yu Tseng, Jia-Bin Huang, Maneesh Kumar Singh, and Ming-Hsuan Yang. Diverse image-to-image translation via disentangled representations. In *European Conference on Computer Vision*, 2018. 2, 3

[10] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Transactions on Image Processing*, 26(7):3142–3155, July 2017. 3

[11] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. *CoRR*, abs/1703.10593, 2017. 2

[12] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networkss. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, 2017. 2, 3