Cloud-GAN: Cloud Removal for Sentinel-2 Imagery Using a Cyclic Consistent Generative Adversarial Network

Praveer Singh, Nikos Komodakis

CLOUD-GAN: CLOUD REMOVAL FOR SENTINEL-2 IMAGERY USING A CYCLIC CONSISTENT GENERATIVE ADVERSARIAL NETWORKS

Praveer Singh and Nikos Komodakis

École des Ponts ParisTech & Université Paris Est, France
Praveer.Singh@enpc.fr

ABSTRACT

Cloud cover is a serious impediment in land surface analysis from Remote Sensing images either causing complete obstruction (thick clouds) with loss of information or blurry effects when being semi-transparent (thin clouds). While thick clouds require complete pixel replacement, thin cloud removal is fairly challenging as the atmospheric and land-cover information is inter-twined. In this paper, we address this problem and propose a Cloud-GAN to learn the mapping between cloudy images and cloud-free images. The adversarial loss in the proposed method constrains the distribution of generated images to be close enough to the underlying distribution of the non-cloudy images. An additional cycle consistency loss is used to further restrain the generator to predict cloud-free images only of the same scene as reflected in the cloudy images. Our method not only rejects the necessity of any paired (cloud/cloud-free) training dataset but also avoids the need of any additional (expensive) spectral source of information such as Synthetic Aperture Radar imagery which is cloud penetrable. Lastly, we demonstrate the efficacy of our technique by training on an openly available and fairly new Sentinel-2 Imagery dataset consisting of real clouds. We also show significant improvement in PSNR values after removing clouds on synthetic images thus validating the competency of our methodology.

Index Terms— Cloud Removal, Generative Adversarial Networks, Deep Learning, Sentinel-2 Imagery.

1. INTRODUCTION

Remote Sensing (RS) imagery is pivotal for varied challenging tasks such as recognizing footprints of buildings [1], detecting changes in temporarily apart scenes [2] or semantic segmentation in aerial scenes [3]. Such images are often plagued by films of clouds that partially or completely obstruct the scene. This can be quite annoying for RS experts, especially while observing a city like Paris which witnesses cloudy weather for a major part of the year. Thus, it clearly necessitates the requirement for an automatic technique that detects and removes the cloudy regions in a scene and replaces them with a neat in-painting of the underlying scene.

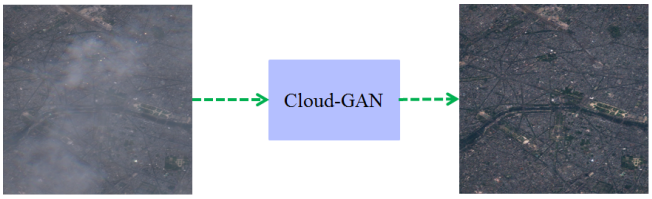
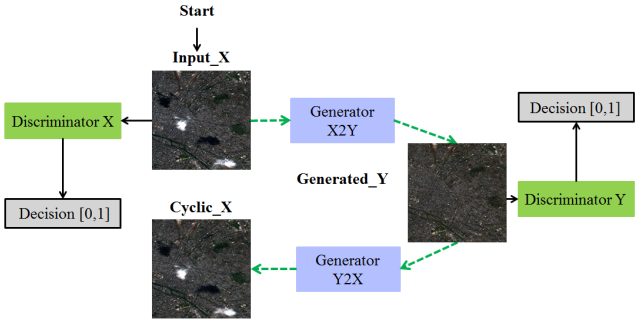


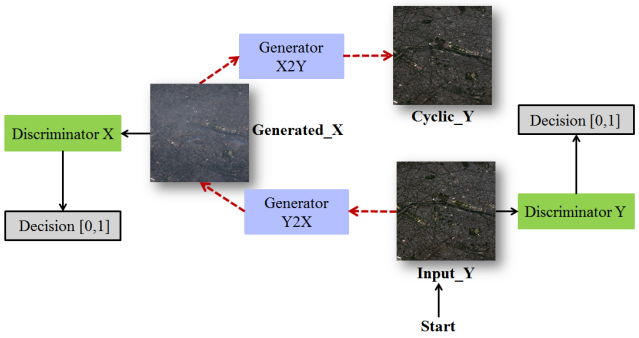
Fig. 1: Cloud-GAN can effectively remove clouds from thin cloudy satellite imagery without supervision using ground truth

Predicting a scene beneath a cloud is an under-constrained problem and unless we have some prior information, it is largely quite complex to replace clouds with correct underlying details. A way out has been by using multi-temporal images of the same region as done by [4] through a Multi-Temporal Dictionary Learning. [5] used Synthetic Aperture Radar (SAR) Imagery owing to the fact that it can easily penetrate through the clouds. However, SAR imagery is difficult to interpret and have a lower spatial resolution compared to RGB imagery. Additionally, [6, 7] studied the thin cloud removal problem in the literature. However, most of these approaches were based on conventional hand crafted methods and are limited in terms of performance.

Generative Adversarial Networks (GANs) [8] have gained immense popularity owing to their remarkable capability in modeling the mapping function between input and output images belonging to target domains. Using an adversarial loss, the GAN's can be trained to produce fake images which are indistinguishable from the real images of target domain. [9] used McGANs to predict cloud-free RGB images as well as cloud masks from the input cloudy image. The authors trained the model using pair of cloud-free image and synthetically produced cloudy images (by adding Perlin noise to the original RGB images). Additionally, they also utilize Near-Infrared (NIR) imagery, which is closer to visible range and possess partial cloud penetration capabilities. However, such kind of synthetic clouds produced using Perlin noise, are not realistic and significantly different from actual clouds seen in visible light images. Nevertheless, composing a dataset of real clouds and their cloud-free counterparts is quite a herculean task.



(a) Forward cycle consistency loss $x \longrightarrow G(x) \longrightarrow F(G(x)) \approx x$



(b) Backward cycle consistency loss $y \longrightarrow F(y) \longrightarrow G(F(y)) \approx y$

Fig. 2: Network Architecture: $Generator_{X2Y}$ and $Generator_{Y2X}$ represent mapping function $G : X \longrightarrow Y$ & $F : Y \longrightarrow X$ respectively. Discriminator X and Discriminator Y represent D_X and D_Y respectively.

We overcome this hurdle, by improvising upon a novel technique, which together with the adversarial loss from traditional GANs, employs a more recent cycle consistent loss [10], to convert thin cloudy images to cloud-free RGB images. Having a cycle consistency loss constrains the problem, such that if an image is transformed from input domain to target and then back to the input domain, it should look alike to the original image. An additional advantage of our method is that it absolves us from the requirement of an explicit paired cloudy/cloud-free dataset. Moreover, our methodology doesn't require any sort of cloud-penetration sources of imagery such as SAR or NIR. We simply utilize visible range imagery from a fairly new open source dataset (Sentinel-2) to report impressive results (figure 1 and 3) clearly showcasing the efficacy of our results. We also report quantitative results on synthetically generated cloudy images, showcasing a significant improvement in PSNR values in Figure 5.

2. PROPOSED FRAMEWORK AND METHODOLOGY

Our proposed framework broadly consists of modeling a mapping function that can translate cloudy images into cloud-free visible range images.

Assuming cloudy images belong to domain X and cloud-free images to domain Y, we define two mapping functions $G : X \longrightarrow Y$ and $F : Y \longrightarrow X$ which are modeled using two generator networks, $Generator_{X2Y}$ and $Generator_{Y2X}$, respectively, as illustrated in the figure 2. Training dataset is composed of $\{x_i\}_{i=1}^{N}$ and $\{y_i\}_{j=1}^{M}$ samples where $x_i \in X$ and $y_i \in Y$ and $p_{data}(x)$ and $p_{data}(y)$ are their data distributions respectively. Also, we have two Discriminator networks, Discriminator X (D_X) and Discriminator Y (D_Y) that distinguish between real data (x and y) and adversaries generated by the two generators i.e. G(x) and F(y).

Hence, in a way the generators and the discriminators compete with each other until they reach to a nash equilibrium. At this instant, the generator is effectively able to match the distribution of generated images (G(x) & G(y)) to the distribution of targeted images ($p_{data}(y)$&$p_{data}(x)$ respectively) and thus, fool the discriminator. We specifically utilize Least Square GAN's (LSGAN's), which have shown [11] to generate higher quality images with a much more stable learning process compared to regular GANs. The adversarial objective function is formulated as:

$$\begin{aligned}\min_{D_Y} L_{LSGAN}(D_Y, X, Y) = E_{y\sim p_{data}(y)}[(D(y)-1)^2] + \\ E_{x\sim p_{data}(x)}[D(G(x))^2] \\ \min_{G} L_{LSGAN}(G, X, Y) = E_{x\sim p_{data}(x)}[(D(G(x))-1)^2]\end{aligned} \tag{1}$$

Similarly, for mapping function F and discriminator D_X, we have objective given by:

$$\min_{D_X} \mathcal{L}_{LSGAN}(D_X, Y, X), \min_{F} \mathcal{L}_{LSGAN}(F, Y, X) \tag{2}$$

Adversarial objective alone is quite under-constrained as the same input can be mapped to any random permutation of target images. Thus, we additionally use a cycle consistency loss which restrains the generator to map a given input x_i to a desired output y_i. In Figure 2 (a), the loss constrains the generated Cyclic X to match Input X i.e. $x \longrightarrow G(x) \longrightarrow F(G(x)) \approx x$, called as forward consistency. Similarly, for backward consistency, $y \longrightarrow F(y) \longrightarrow G(F(y)) \approx y$. We formulate the cycle consistency objective as:

$$\begin{aligned}\min_{G,F} \mathcal{L}_{cyc}(G, F) = E_{x\sim p_{data}(x)}[\| (F(G(x)) - x\|_1] + \\ E_{y\sim p_{data}(y)}[\| (G(F(y)) - y\|_1]\end{aligned} \tag{3}$$

Combining the Generator objectives from equations 1 and 2 and cyclic-consistency loss, our final Generator objective is:

$$\begin{aligned}\min_{G,F} \mathcal{L}_{Gen}(G, F) = \mathcal{L}_{LSGAN}(G, X, Y) + \\ \mathcal{L}_{LSGAN}(F, Y, X) + \lambda \mathcal{L}_{cyc}(G, F)\end{aligned} \tag{4}$$

where λ is a regularizing factor that weights the cyclic term with respect to Generative term of the adversarial objective function.

3. TRAINING AND IMPLEMENTATION DETAILS

3.1. Dataset

Our dataset is composed of high resolution Level-1C Sentinel-2 imagery ranging between the year 2015 till 2017. Sentinel-2 is a multi-spectral dataset, with each spectral band is stored as a separate image [12]. For our experiments, we choose images only from visible bands i.e. Blue (B2), Green (B3), Red (B4) all of which have 10 meters of spatial resolution.

Most of our cloud-free images are selected with 0-5% cover while for cloudy images we chose range anywhere between 10 to 100. All the images are downloaded over the Paris region as it is easy to get quite a range of cloudy images. We choose 20 cloudy and 13 cloudless images for training. We then extract 512×512 patches from these images. After filtering of unwanted ones, a total of 1677 patches for each cloud and cloud-free dataset were extracted while for testing we had 837 patches. For computational efficiency in training, we resize them to 256×256.

3.2. Network Architectures

We imbibe the architecture and the naming convention similar to what have been used by [10]. Generator architecture, uses 6 blocks for 128×128 training images and 9 blocks for 256×256 or higher resolution images. Additionally, a reflection pad is imbibed to avoid artifacts. The Discriminator architecture consists of a 70×70 PatchGAN [8], classifying 70×70 patches as real or fake data. Thus, it can effectively be applied to any input size image and has lesser number of parameters.

3.3. Training

Initialization of weights was done through a Gaussian distribution with mean 0 and standard deviation 0.02. Optimization was carried out using ADAM [13], with a batch size of 1 and $\lambda = 10$ for all experiments. We perform training from scratch using a learning rate of 0.0002 up-to 200 epochs. The learning rate was kept constant for the first 100 epochs after which it linearly decays to zero until the last epoch. Also, as illustrated in [10] model oscillations are avoided by using a history of generated images (50) rather than only one.

4. RESULTS AND DISCUSSIONS

We present the results obtained using Cloud-GAN in Figure 3 (for real clouds) and 4 (for synthetic clouds). Without using any corresponding Cloud-Free pair for a cloudy image, our Cloud-GAN efficiently removes thin clouds spread throughout a scene, as shown in row III, IV, V in Figure 3. More interestingly, it effectively detects small cloudy patches and replaces them with the underlying ground details, as depicted in row I and II in Figure 3. Cloud-GAN interestingly is able to retain finer details like patches of urban settlements, river, fields (row IV) while getting rid of the cloudy film. In some cases *e.g.*, row II, the generated image from our method is more natural and visually more pleasing than the original image which is a byproduct of our method.

We cannot report any quantitative results on the real dataset since we lack paired cloudy-cloud-free images. However, to compensate for that, we report results on 5 synthetic scenes, composed by addition of Perlin noise to cloud free images. We provide the corresponding PSNR results in Figure 5. We observe that even though we trained our model on real dataset, our model substantially outperforms on all synthetic scenes by a significant margin. Figure 4 shows two of these test scenes. Note that we do not provide comparison with [9], due to the unavailability of their code and dataset.

We additionally show some special instances of thick cloud (Figure 6) where our model fails to yield credible results. In Figure 6 row I, we see that the model contends by generating an over-smoothed image when the clouds are too opaque. In Figure 6 row II, the model fails completely to produce an image as the clouds have occupied most of the visible area. One of the reasons can be that the network finds no closest sample in the target dataset and hence predicts a spatially smooth region under the cloud or some random noise. The only way to solve this is by the addition of an extra source such as SAR images which can penetrate through these clouds and give us details of the underlying ground details.

5. CONCLUSIONS

We have proposed a novel technique to remove thin clouds from Sentinel-2 imagery. Our cloud removal technique is employed without using any explicit dataset of paired Cloudy/Cloud-Free images as performed in the past. Our model nullifies the requirement of creating and training on synthetic dataset which is not truly realistic. Our input sources are purely visible range images without any prerequisite for SAR or other cloud-penetration sources. However, presence of thick clouds do necessitate need of an additional high wavelength imagery to gain some knowledge about the underlying ground information which we leave as a topic of future research. Lastly, we report PSNR results on some synthetic test scenes where we see significant improvement in performance thus validating the efficacy of our results.

6. REFERENCES

[1] Emmanuel Maggiori, Yuliya Tarabalka, Guillaume Charpiat, and Pierre Alliez, "Can semantic labeling methods generalize to any city? the inria aerial image labeling benchmark," in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. IEEE, 2017.

[2] H. Lyu, H. Lu, and L. Mou, "Learning a transferable change rule from a recurrent neural network for land cover change detection," *Remote Sensing*, vol. 8, no. 6, 2016.

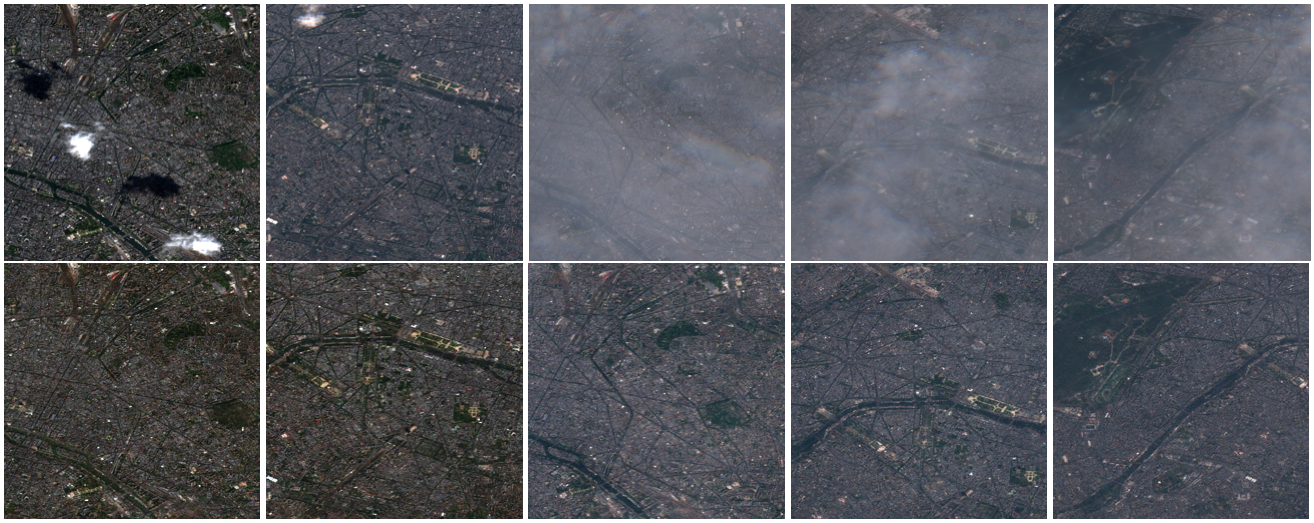


Fig. 3: *Qualitative Results* on Real Cloud Dataset. Cloudy images are shown in **Row I** and Cloud-GAN generated cloud-free images in **Row II**.

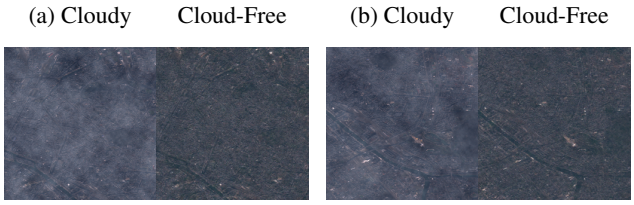


Fig. 4: Sample images from Synthetic dataset where (a) Scene-5 (b) Scene-4

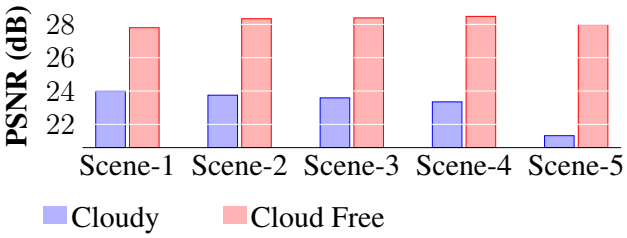


Fig. 5: Quantitative results on Synthetic Scenes

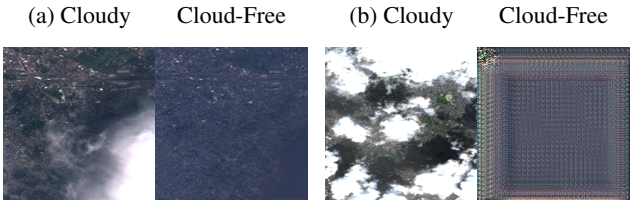


Fig. 6: Failure Cases: Over-smoothening or completely fail to produce images for overly clouded images

[3] N. Audebert, B. Le Saux, and S. Lefvre, "How useful is region-based classification of remote sensing images in a deep learning framework?," in *IGARSS*, July 2016, pp. 5091–5094.

[4] M. Xu, X. Jia, M. Pickering, and A. J. Plaza, "Cloud removal based on sparse representation via multitemporal dictionary learning," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 5, pp. 2998–3006, May 2016.

[5] B. Huang, Y. Li, X. Han, Y. Cui, W. Li, and R. Li, "Cloud removal from optical satellite imagery with sar imagery using sparse representation," *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no. 5, pp. 1046–1050, May 2015.

[6] Huanfeng Shen, Huifang Li, Yan Qian, Liangpei Zhang, and Qiangqiang Yuan, "An effective thin cloud removal procedure for visible remote sensing images," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 96, pp. 224–235, 2014.

[7] M. Xu, M. Pickering, A. J. Plaza, and X. Jia, "Thin cloud removal based on signal transmission principles and spectral mixture analysis," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 3, pp. 1659–1669, March 2016.

[8] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros, "Image-to-image translation with conditional adversarial networks," *CVPR*, 2017.

[9] K. Enomoto, K. Sakurada, W. Wang, H. Fukui, M. Matsuoka, R. Nakamura, and N. Kawaguchi, "Filmy cloud removal on satellite imagery with multispectral conditional generative adversarial nets," in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, July 2017, pp. 1533–1541.

[10] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," *arXiv preprint arXiv:1703.10593*, 2017.

[11] Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, Zhen Wang, and Stephen Paul Smolley, "Least squares generative adversarial networks," in *2017 IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2017, pp. 2813–2821.

[12] Onera, "Medusa toolbox," `http://w3.onera.fr/medusa/downloading`, 2017.

[13] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014.