Aerial GANeration: Towards Realistic Data Augmentation Using Conditional GANs

Stefan Milz, Tobias Rüdiger, Sebastian Süss

Spleenlab
{stefan.milz,tobias.ruediger,sebastian.suess}@spleenlab.com

Abstract. Environmental perception for autonomous aerial vehicles is a rising field. Recent years have shown a strong increase of performance in terms of accuracy and efficiency with the aid of convolutional neural networks. Thus, the community has established data sets for benchmarking several kinds of algorithms. However, public data is rare for multi-sensor approaches or either not large enough to train very accurate algorithms. For this reason, we propose a method to generate multi-sensor data sets using realistic data augmentation based on conditional generative adversarial networks (cGAN). cGANs have shown impressive results for image to image translation. We use this principle for sensor simulation. Hence, there is no need for expensive and complex 3D engines. Our method encodes ground truth data, e.g semantics or object boxes that could be drawn randomly, in the conditional image to generate realistic consistent sensor data. Our method is proven for aerial object detection and semantic segmentation on visual data, such as 3D Lidar reconstruction using the ISPRS and DOTA data set. We demonstrate qualitative accuracy improvements for state-of-the-art object detection (YOLO) using our augmentation technique.

Keywords: Conditional GANs, Sensor Fusion, Aerial Perception, Object Detection, Semantic Segmentation, 3D Reconstruction

1 Introduction

Aerial perception is a rising field for autonomous vehicles. Especially algorithms based on large data sets have shown accurate results in recent years. Despite all advances, we believe that fully autonomous navigation in arbitrarily complex environments is still far away, especially for automated aerial transportation including all safety aspects. The reasons for that are manifold. On one hand, highly accurate algorithms on dedicated hardware with real-time capabilities are needed for perception. On the other hand, almost all leading state-of-the-art perception (see the DOTA leader board [2]) algorithms are based on deep learning that require individually designed large scaled data sets for training. Within this paper, we want to target the second issue and propose a new method for realistic data augmentation in the domain of aerial perception using cGANs. We

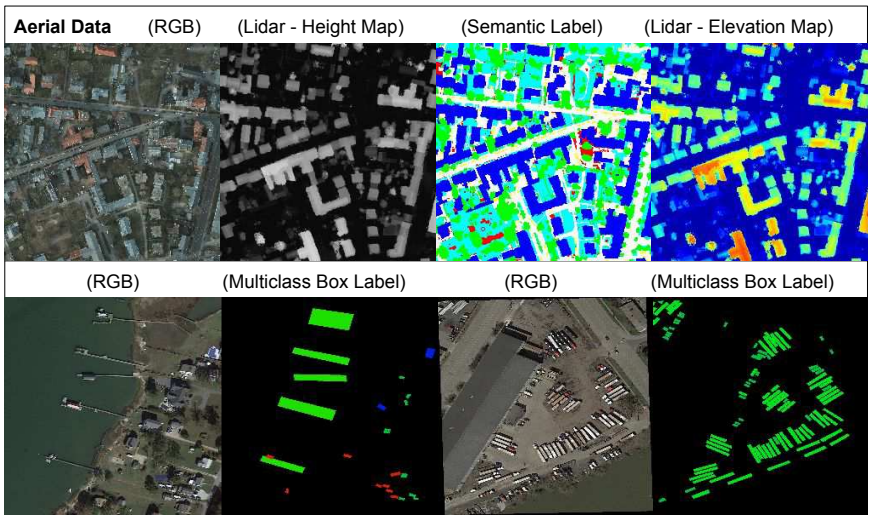

Fig. 1. Exemplary aerial data used for GANeration. The upper row shows samples given by the ISPRS Dataset (Potsdam) [1] representing RGB, Lidar and Semantic Segmentation Labels. The lower row represents two samples from the DOTA [2] data set with multi-class object boxes (colorized by classes) and spatially encoded inside an RGB image. Additionally, the visual camera RGB image is shown.

evaluate qualitatively the data generation for three different tasks: object detection, semantic segmentation and 3D reconstruction based on two sensor types: Cameras and Lidar (ISPRS [1] and DOTA[2]) (see Fig.1). Additionally, we show significant accuracy improvements for the YOLOv2 [3] object detector using a small subset of the DOTA training base compared to the same detector trained on an augmented extension set using our proposed method. The latter yields much better accuracy without any change of architecture, purely influenced by the GANerated training set.

1.1 Contribution

We present the first approach for synthetic aerial data generation without the need for a complicated 3D engine or any exhausting preprocessing. The proposed method is independent from the desired perception task. This is evaluated by several qualitative experiments, like object detection, semantic segmentation or 3D reconstruction. The method strongly improves the accuracy of a perception algorithm that is exemplary demonstrated by an aerial object detection using YOLOv2. On top, the method can produce different kinds of sensor data, like camera images or Lidar point clouds. The basic idea is the usage of a cGAN, where the desired ground truth is used as conditional input. Here, we encode

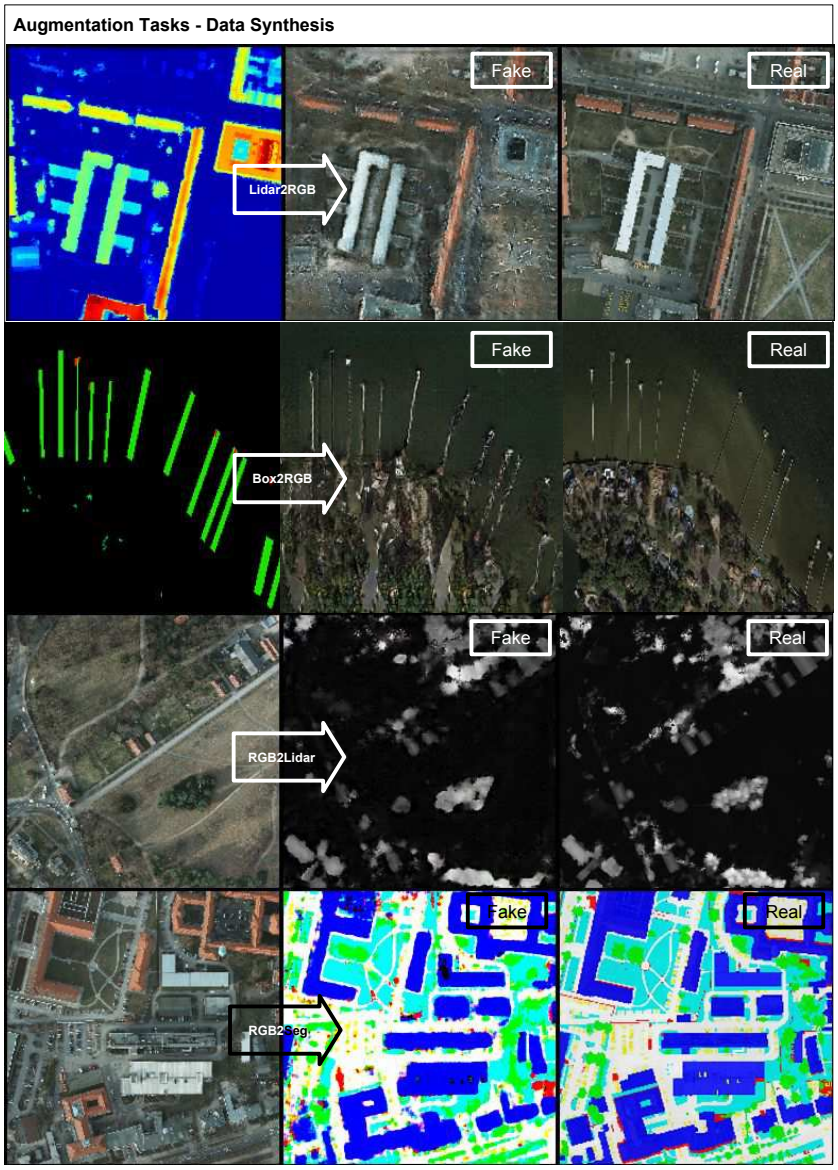

Fig. 2. Exemplary augmentation tasks for Aerial GANeration: Our approach using a cycle GAN could be used generically. Neither sensor nor data representation does matter. Any kind of data synthesis is possible. 1. The image synthesis based on a Lidar scan. 2. The generation of an RGB image based of ground truth 2D bounding boxes. 3. The 3D reconstruction (height map) of an RGB image. 4. The semantic segmentation of an aerial image.

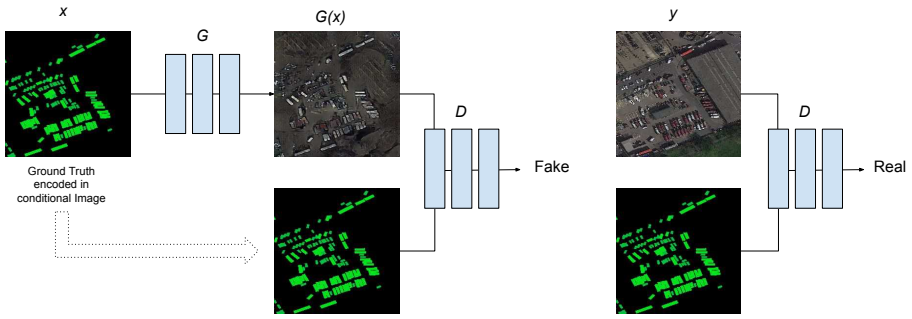

Fig. 3. Ground truth encoding in conditional images. This figure shows the typical structure of a cGAN playing the minimax game. A generator G is used to create a fake image $G(x)$ based on the conditional image x, e.g. Pix2Pix[4]. The discriminator D tries to distinguish between real $D(y)$ and fake images $D(x)$. Aerial GANeration encodes ground truth data in the conditional image x to produce realistic sensor data. The basic idea is to encode ground truth images that are easy to collect or can be sampled automatically, e.g. 2D bounding boxes could be drawn randomly with color classes in an image x to generate a realistic accompanied image $G(x)$.

the condition as an image pair, i.e. the algorithm works even well vice versa (see Fig.1).

2 Conditional GANs

2.1 Related Work

In contrast to predictive neural networks that are used for classification and regression purposes, generative networks are not as manifold. The reason is a much more challenging training process. Hence, the use and spreading have started just some years ago, when Goodfellow at al. [5] presented their ground breaking publication of GANs in 2014. Although other methods like deep belief networks [6] or generative autoencoders [7] exist, GANs have developed to the most common generative neural networks.

Basically GANs use a random noise vector to generate data. As the applications for a totally random data generation are very limited, Goodfellow at al.[5] have already described methods for adding parameter to the input signal that allow an adaptation of the network output. GANs that apply this method by an additional conditional input are called cGANs.

cGANs have been widely used to produce realistic data out of the latent space initiated on a conditional vector. Research and concurrent work has been done on discrete labels [8], text and images ([4], [9]). The latter has been very popular in the domain of image to image translation. The idea behind a conditional GAN

for image translation is to encode the condition inside an image to generate accompanied data. This is also known as per-pixel classification or regression.

We adapt this approach to augment datasets for aerial use cases with the aid of encoding easy to generate ground truth inside the conditional image. Accompanied sensor data, e.g. RGB images, are generated by the generator G, whereas the discriminator D decides weather an image is fake or not (see Fig. 3). Similar to Isola et. al [4] we use a Unet-Architecture [10] for the generator and PatchGAN for the discriminator [11].

2.2 Image to Image Translation

In general GANs were developed to create an image y based on a random noise vector z. $G : z \rightarrow y$ [4]. In contrast a cGAN produces an image by using a noise vector z and a conditional vector c. $G : [c, z] \rightarrow y$. In terms of Image to Image translation c is an input image x. Hence, we use the following convention for mapping $G : [x, z] \rightarrow y$ (see Fig.3).

2.3 Objective

The objective of a basic GAN can be described by an additive combination of the loss of the discriminative network D and the generative network G. In order to create more and more realistic data, the loss is reduced by training G, whereas a training step of D results in an increase of the loss ideally. Consequently, we can describe both parts of the loss as follows:

$$L_{GAN}(G, D) = \mathbb{E}_y\{\log(D(y))\} + \mathbb{E}_{x,z}\{\log(1 - D(G(x, z))\} \tag{1}$$

The loss in this form is suitable for generating totally random output images, as the discriminative network does not take the conditional input x into account. As our purpose is to enlarge data sets by generating data based on a ground truth image, we need to extend the loss in a way that adds a considering of the conditional input in the network D:

$$L_{cGAN}(G, D) = \mathbb{E}_{x,y}\{\log(D(x, y))\} + \mathbb{E}_{x,z}\{\log(1 - D(x, G(x, z))\} \tag{2}$$

Due to the findings in [4], we add a weighted L1 distance to the loss of the conditional network. The overall loss of our network setup can be written as:

$$L = L_{cGAN}(G, D) + \lambda \cdot \mathbb{E}_{x,y,z}\{||y - G(x, z)||_1\} \tag{3}$$

According to the recommendations in [4], we did not use a dedicated noise vector as input image. As the network tends to ignore the noise vector input, this approach would lead to unneeded computational effort and reduced computational efficiency. However, we applied the noise in some of the generator network layers to achieve some kind of randomness in the network output.

From Fig.2, it can be seen that the noise we have added to the generator network has no large effect on the network output. Although, noise is added, the

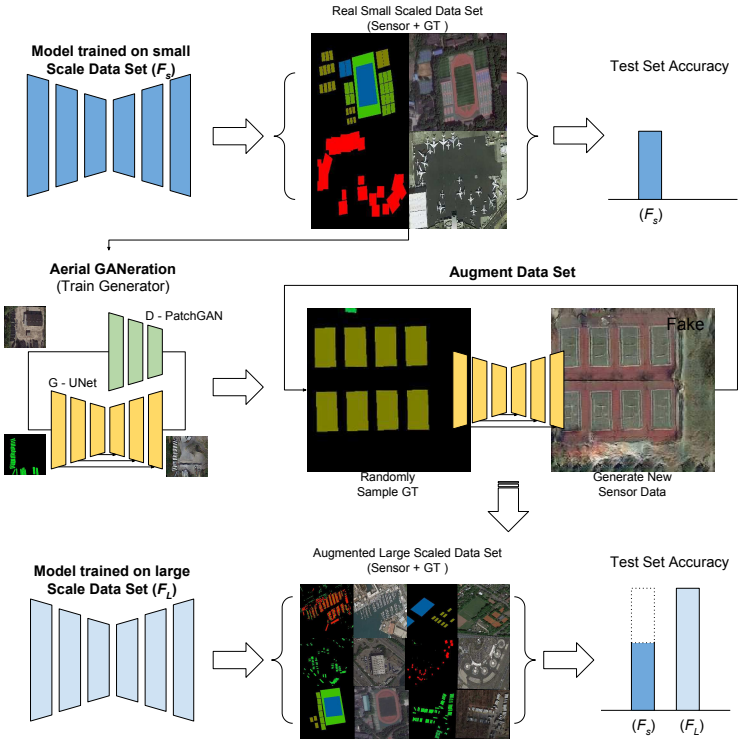

Fig. 4. Augmentation Strategy of aerial GANeration. Our approach improves the quality and accuracy of state-of-the-art task related neural networks (e.g. object detector) by realistic augmentation using conditional GANs. The upper row shows a CNN trained on a small dataset with a sufficient performance F_S. The middle row describes the augmentation strategy. The lower row outlines the new training on the augmented dataset with a strong accuracy improvement: $F_L >> F_S$. Note: The test-set does not include any augmented data.

fake images do not differ much from the real ones. Hence, the generated images are similar but not equal to the real images. Nevertheless, we can show in the following that the achieved differences in the output images are sufficient to add further diversity to a data set so that the performance of predictive networks trained on such a data set, that has been enlarged by cGANs, is improved. We will show this by applying training YOLO on both an extended and an unextended data set and evaluate the prediction performance.

2.4 Augmentation Strategy

The basic idea of our augmentation strategy is to bypass the expensive methods for data synthesis, e.g. the simulation of a realistic 3D engine. We focus on "easy-

to-get" ground truth for what we generate input data. Our proposed model therefore consists of four steps (see all listed steps in Fig.4):

1. Get an annotated small scale data set (RGB + bounding boxes)
2. Train the aerial GANerator using a cGAN
3. Augment the small scale data set to a large data set by sampling ground truth randomly → Encode them inside the conditional image
4. Improve the task (e.g. object detector) related deep learning approach via re-training on a large training base

3 Experiments

Our ablation study is divided into a quantitative and a qualitative part. First we present quantitative results on any kind of data generation. Second, we show significant improvements qualitatively comparing the same state-of-the-art object detector YOLO trained on a base and on an extended dataset using our augmentation method. In general, evaluating the quality of synthesized images is an open and difficult problem. Consequently, we explore in our quantitative study visual problems like RGB image creation or 3D reconstruction (root mean square error assessment), such as visual tasks, like semantic segmentation (intersection over union assessment). The study includes the following applications:

- **Visual qualitative results**:
  - Aerial RGB ↔ Semantic segmentation on ISPRS [1][1]
  - Aerial RGB ↔ Lidar height-map on ISPRS [1]
  - Aerial RGB ↔ Lidar elevation-map on ISPRS [1]
  - 2D multi-class box labels → RGB on DOTA [2][2]
- **Quantitative detection results**:
  - 2D multi-class box labels → RGB on DOTA [2] and training on augmented dataset using YOLOv2[3]

We tested our proposed method on DOTA[2] and ISPRS[1]. Especially our use cases for ISPRS are based on the *Potsdam* part, which contains RGB, semantic segmentation and Lidar. For exploring the DOTA data, we split the available training data set containing 1411 samples with accompanied ground truth boxes with 15 different classes into 706 training and 705 test samples. The ISPRS data set that contains 40 images was split into 37 training and 3 test images that are mainly used to explore visual results. The model itself is based on Isola et. al [4] for all evaluation studies, with a GGAN loss function, 200 epochs, resized image crops to 256 × 256 pixels and batch normalization.

RGB to Semantic Segmentation and Vice Versa The results for RGB to semantic translation are shown in Fig.5 with 6 color classes: Impervious surfaces

[1] ISPRS - Part2 → Potsdam
[2] DOTA - Resized to image size of 256x256

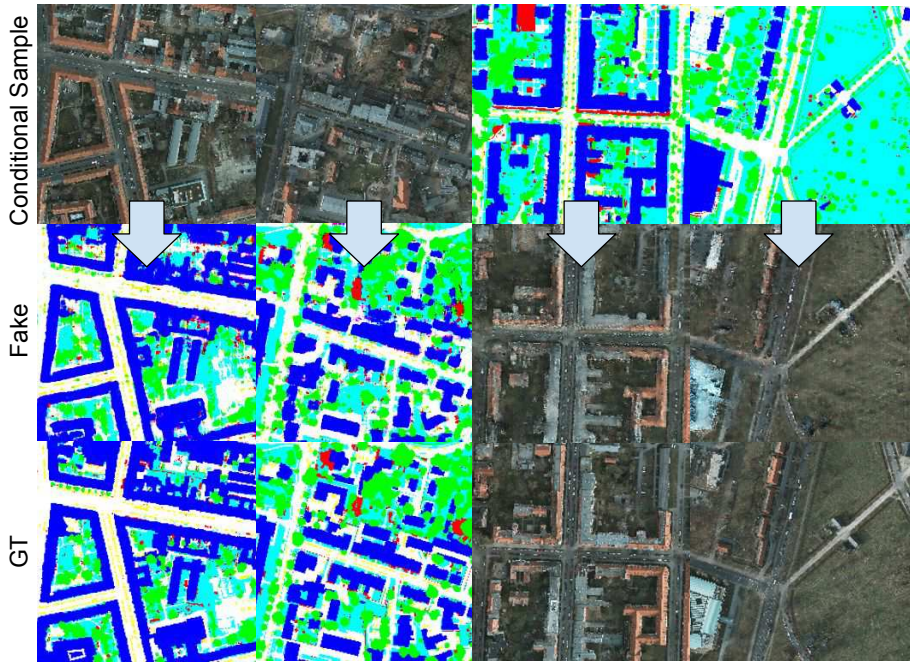

Fig. 5. Results of aerial GANeration for semantic segmentation (left) and semantic to RGB translation (right). The results are based on our split using the ISPRS[1] test set (see section experiments: RGB to Semantic Segmentation)

(white), building (blue), low vegetation (bright blue), tree (green), car (yellow), clutter/background (red). The figure shows the results of the test set. From a visual point of view both cases, i.e. image to segmentation and segmentation to image, seem to be promising. Additionally, we underline our visual results with the values for intersection over union (IOU) [4] on the test set for the segmentation task. Although the test set is very small, the metrics we yielded (Tab.1) are state-of-the-art.

Table 1. IoU for the aerial GANeration approach in the domain of image to semantic segmentation translation (ISPRS dataset: 37 training images, 3 test images)

Classes	**IoU Aerial GAN**
Impervious surfaces	79.4%
Building	87.1%
Low vegetation	67.3%
Tree	70.3%
Car	24.1%
Clutter/background	30.7%
Mean IoU	**59.8%**

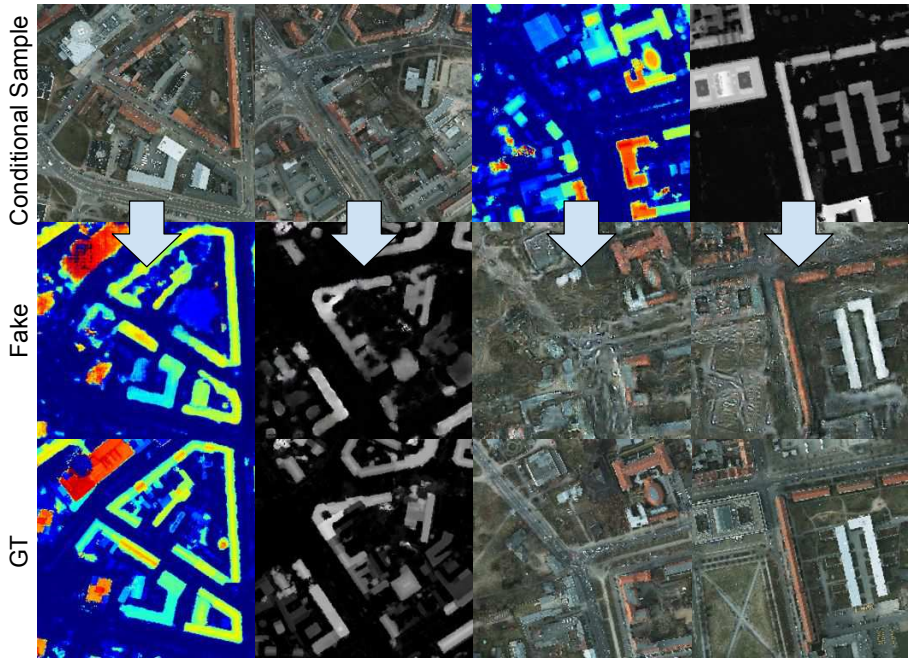

Fig. 6. Results of Aerial GANeration for 3D reconstruction (left) and Lidar to RGB translation (right). The results are based on our split using the ISPRS[1] test set (see section experiments: RGB to Lidar)

RGB to 3D Lidar-Reconstruction and Vice Versa Fig. 6 shows the qualitative results of our Lidar data 3D generation and the Lidar to RGB translation. Both use cases are either realized via height or colorized elevation map encoding. Again, our experiments show promising results.

To verify the visual findings approximately, we calculated the root mean square error (RMSE) on pixel level as relative RMSE [4] for both encodings using our test set in the domain of RGB to Lidar translation. The results are shown in Tab.2. To our surprise, the results for the height map are much more accurate than those for the elevation map. However, we explain this with the quantization of the much smaller prediction range (8 bit vs. 24 bit) and the random behavior of the too small selected test set.

Table 2. Relative Root Mean Squared Error on pixel level for 3D reconstruction using aerial GANeration (ISPRS dataset: 37 training images, 3 test images)

Classes	**rRMSE Aerial GAN**
Lidar Height Map	14.53%
Lidar Elevation Map	21.24%

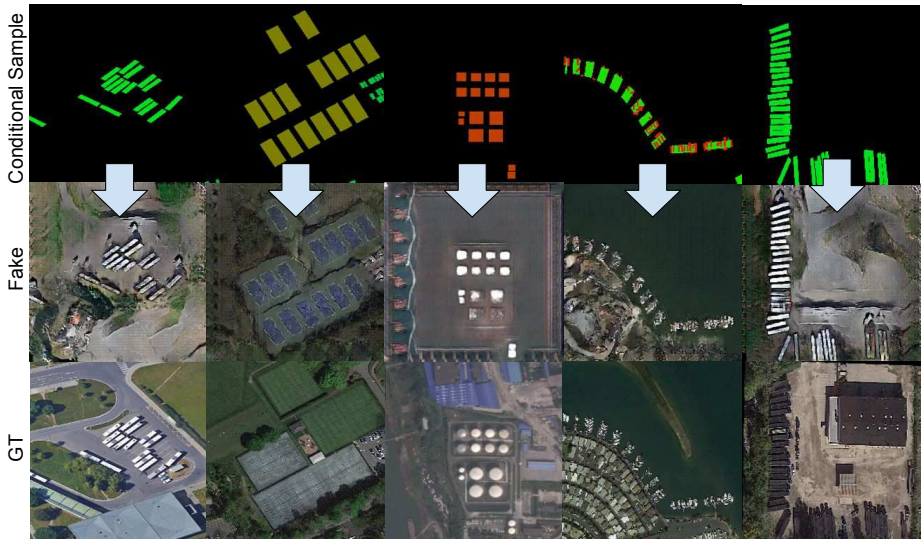

Fig. 7. Results of aerial GANeration for box label to image translation. The results are based on our split using the DOTA[2] test set (see section experiments: Multi-Box Labels to RGB)

Multi-Class Box Labels to RGB The following experiments are based on the DOTA[2] containing 1411 samples (50:50 split) and 15 different classes. Therefore, this experiment has a higher significance than the results on the ISPRS data set. Additionally, the dataset contains different viewpoints. Hence, the model has to learn the scale invariance. At least, we resized all images to an input size of 256×256. Those qualitative results for image predictions based on input boxes are shown in Fig.7. We yield promising results for classes with a less complex structure like *tennis-court*, *large vehicle* or *storage tank*. Due to the scale variance and the low input image size, we observed failure cases for more complex structures Fig.8. Indeed, the model is not feasible to perform the object detection itself, i.e. the inverse translation problem (image to box) Fig.9. The experiment never converged for our setup. We believe, the main reason for this is the extreme viewpoint variance inside the image dataset, which is a typical problem for aerial perception.

Improving YOLO[3] using aerial GANeration Unless weaknesses were observed in the previous section, the full augmentation method was applied to the state-of-the-art object detector YOLO. The concept was validated with the aid of the DOTA training data set for the parallel or horizontal Multi-class object box detection. We use the same split as described previous, i.e. 1411 samples containing 706 training and 705 test cases. Again, we down sample every image

Fig. 8. Failure cases for aerial GANeration. The figure outlines weaknesses of the cGAN to generate complex structures (left two columnes). The middle column shows drawbacks in generating small objects in terms of a far viewpoint.

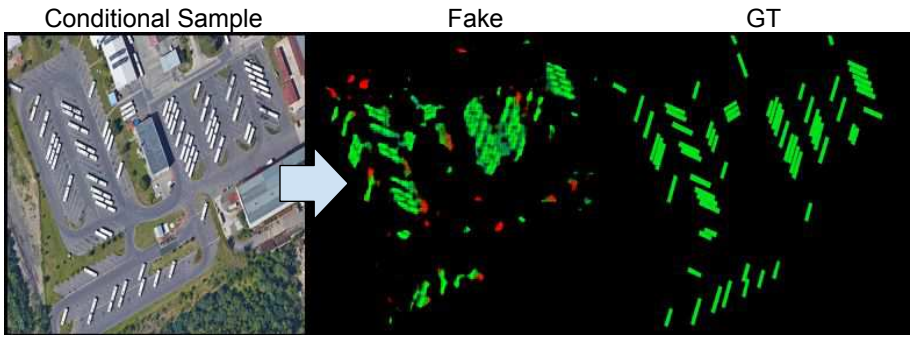

Fig. 9. Aerial GANeration for inverse object box image creation (image2objectboxes), i.e. the object detection itself, where the model never converged in our experiments. The figure shows a sample of the described failure case. It is not possible to extract boxes out of the generated images.

to 256×256 pixels. This drastically affects the results, which are not competitive to the official leader board. However, it simply shows the influence of our model.

The augmentation procedure is divided into four phases (see Fig.4):

1. YOLOv2 (F_s) is trained on the small scale training base
2. The training base is augmented from 706 $\Rightarrow$ 1412 by sampling equally distributed bounding boxes according to the distribution (position, rotation, height, width) inside the dataset using k-means clustering
3. YOLOv2 (F_l) is retrained on the large augmented training set
4. Both models (F_s, F_l) are compared with the aid of the test set (705 samples) in terms of accuracy

We show significant improvements especially for objects with a low complexity, e.g. baseball diamond, ground track field, large vehicle, tennis court or swimming pool. The improvement is not recognizable for complex objects like planes or ships[3]. However, we believe that those results prove the main idea of our concept. An improved architecture may lead to much better results and could be applied to any kind of sensor data generation. This could facilitate data generation for any kind of perception task, especially aerial cognition.

Table 3. Improving YOLOv2 using the Aerial GANeration. Our experiments are validated using the DOTA data set based on our individual split. Bold values emphasize object class specific improvements. The experiment increases performance for simple objects. We used the standard YOLOv2 architecture similarly trained for 10000 iterations. Both experiments run the standard YOLOv2 augmentation strategy ontop. The test-set does not include any augmented data.

Classes	**mAP** F_s	**mAP** F_l
plane	0.66%	0.65%
baseball diamond	0.43%	**0.49**%
bridge	0.16%	**0.18**%
ground track field	0.38%	**0.45**%
small vehicle	0.41%	0.41%
large vehicle	0.54%	**0.58**%
ship	0.51%	0.49%
tennis court	0.61%	**0.66**%
basketball court	0.67%	**0.72**%
storage tank	0.45%	**0.46** %
soccer ball field	0.19%	**0.24**%
roundabout	0.21%	0.20%
harbor	0.39%	0.39%
swimming pool	0.33%	**0.38**%
helicopter	0.46%	0.46%
mAP IoU	0.43%	**0.46%**

[3] Note, the officially published DOTA leader board results are much better due too the higher input image size. For simplicity, we downscale all the images to $256x256$

4 Conclusion

Large scale aerial data sets for deep learning purposes are rare so far. Hence, the development of high performance classification algorithms requires the creation of novel, large scale data sets or the extension of existing data sets. In this paper we treated the second approach of extending current data sets. We addressed this topic by a computational efficient approach. We suggested to use cGANs that do not require complex simulations or 3D engine processing for data generation. We demonstrated the versatility of cGANs by applying them to a couple of different generation problems. This includes generation of semantic segmentation based on RGB images as ground truth and vise versa, of RGB based on Lidar data and of 2D multi-class box based on RGB. The qualitative and quantitative results show the huge potential of cGANs for data generation. By training a YOLO network, we demonstrated the gain that can be achieved by extending training data sets with cGANs.

However, the effect of extending existing small scale data sets with cGANs is limited due to some weaknesses of GANs in general. On the one hand, the low randomness that appears during learning process affects data generation negatively. On the other hand, the performance of cGANs is also depending on the number of training samples. The quality of the generation increases in bigger data sets, so that a chicken-and-egg problem is produced.

Consequently, cGANs are a very effective method to increase classification performance in case of restricted training samples and data set diversity. Nevertheless, for future development of deep learning based algorithms in aerial scenarios, large scale multi sensor data sets are indispensable and need to be addressed in the near future.

4.1 Future Work

The paper has demonstrated the principle possibility, that cGANs help to augment data. However, a detailed ablation study is missing. Moreover, it has to be demonstrated that a real domain translation could be achieved, e.g. Pixels to Point-Clouds or one dimensional signals to pixels. Despite, the authors would like to generate augmented data for corner cases within the aerial vehicle domain, who are impossible to measure, to make aerial perception more explainable and safe.

Acknowledgement

The authors would like to thank their families especially their wifes (Julia, Isabell, Caterina) and children (Til, Liesbeth, Karl, Fritz, Frieda) for their strong mental support.

References

1. Khoshelham, K., Díaz Vilariño, L., Peter, M., Kang, Z., Acharya, D.: The isprs benchmark on indoor modelling. ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences **XLII-2/W7** (2017) 367–372
2. Xia, G., Bai, X., Ding, J., Zhu, Z., Belongie, S.J., Luo, J., Datcu, M., Pelillo, M., Zhang, L.: DOTA: A large-scale dataset for object detection in aerial images. CoRR **abs/1711.10398** (2017)
3. Redmon, J., Farhadi, A.: YOLO9000: better, faster, stronger. CoRR **abs/1612.08242** (2016)
4. Isola, P., Zhu, J., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. CoRR **abs/1611.07004** (2016)
5. Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial networks (2014)
6. Hinton, G.E., Osindero, S., Teh, Y.W.: A fast learning algorithm for deep belief nets. Neural Computation **18**(7) (2006) 1527–1554
7. Tran, N.T., Bui, T.A., Cheung, N.M.: Generative adversarial autoencoder networks (2018)
8. Mirza, M., Osindero, S.: Conditional generative adversarial nets. CoRR **abs/1411.1784** (2014)
9. Zhu, J., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. CoRR **abs/1703.10593** (2017)
10. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. CoRR **abs/1505.04597** (2015)
11. Li, C., Wand, M.: Precomputed real-time texture synthesis with markovian generative adversarial networks. CoRR **abs/1604.04382** (2016)