Image-to-Image Translation for Improvement of Synthetic Thermal Infrared Training Data Using Generative Adversarial Networks

Hanna Hamrell

Master of Science Thesis in Computer Vision
**Image-to-Image Translation for Improvement of Synthetic Thermal Infrared
Training Data Using Generative Adversarial Networks:**

Hanna Hamrell

LiTH-ISY-EX--21/5364--SE

Supervisor: **Gustav Häger**
ISY, Linköpings universitet
Jörgen Karlholm
Totalförsvarets forskningsinstitut (FOI)

Examiner: **Mårten Wadenbäck**
ISY, Linköpings universitet

*Computer Vision Laboratory
Department of Electrical Engineering
Linköping University
SE-581 83 Linköping, Sweden*

Copyright © 2021 Hanna Hamrell

Abstract

Training data is an essential ingredient within supervised learning, yet time consuming, expensive and for some applications impossible to retrieve. Thus it is of interest to use synthetic training data. However, the domain shift of synthetic data makes it challenging to obtain good results when used as training data for deep learning models. It is therefore of interest to refine synthetic data, e.g. using image-to-image translation, to improve results. The aim of this work is to compare different methods to do image-to-image translation of synthetic training data of thermal IR-images using GANs. Translation is done both using synthetic thermal IR-images alone, as well as including pixelwise depth and/or semantic information. To evaluate, a new measure based on the Fréchet Inception Distance, adapted to work for thermal IR-images is proposed. The results show that the model trained using IR-images alone translates the generated images closest to the domain of authentic thermal IR-images. The training where IR-images are complemented by corresponding pixelwise depth data performs second best. However, given more training time, inclusion of depth data has the potential to outperform training with IR data alone. This gives a valuable insight on how to best translate images from the domain of synthetic IR images to that of authentic IR images, which is vital for quick and low cost generation of training data for deep learning models.

Acknowledgements

First I would like to give special thanks to FOI for providing training data and great computational resources. Secondly I would like to thank my supervisors Jörgen Karlholm and Gustav Häger, for long valuable discussions and great knowledge within the field.

Linköping, February 2021
Hanna Hamrell

Contents

Notation	ix
1 Introduction	1
1.1 Motivation	1
1.2 Purpose	1
1.3 Problem Formulation	2
1.4 Limitations	2
2 Related Work	3
2.1 Image-to-image translation	3
2.2 Generative Adversarial Networks	4
2.3 Evaluation of GANs	5
2.4 Monocular Image Depth Estimation	6
2.5 Semantic Image Segmentation	6
3 Method	7
3.1 The Data	7
3.2 Implementation	9
3.2.1 Network Architecture	9
3.2.2 Training Details	14
3.3 Objective	15
3.4 Evaluation	16
3.5 Hardware	18
4 Results	21
5 Discussion	35
5.1 Results	35
5.2 Method	35
5.3 Future Work	36
6 Conclusion	37
Bibliography	39

Notation

MATHEMATICAL NOTATION

Notation	Meaning
$\mathbb{E}$	Expectation
$\mathcal{L}_{\text{GAN}}$	Adversarial loss
$\mathcal{L}_{\text{cyc}}$	Cycle consistency loss
$\mathcal{L}_{\text{identity}}$	Identity loss
$\mathcal{L}_{\text{semantic}}$	Semantic loss
$\mathcal{L}_{\text{depth}}$	Depth loss
X	The domain of synthetic data, including thermal IR images and if specified also corresponding depth and semantic information
Y	The domain of authentic thermal IR images
x	Synthetic data sample
y	Authentic data sample
G	Generator model mapping from domain X to domain Y
F	Generator model mapping from domain Y to domain X
D_Y	Discriminator which aims to distinguish between real authentic data and generated authentic data
D_X	Discriminator which aims to distinguish between real synthetic data and generated synthetic data
$p_X(x)$	Statistical distribution of the stochastic variable x
$p_Y(y)$	Statistical distribution of the stochastic variable y
$\ \cdot\ _1$	$\mathcal{L}_1$ -norm

ABBREVIATIONS

Abbreviation	Meaning
ARD	Automatic relevance determination
CPU	Central processing unit
DNN	Deep Neural Network
FD	Frechét Distance
FID	Frechét Inception Distance
FOI	Totalförsvarets forskningsinstitut or Swedish Defence Research Agency
GAN	Generative Adversarial Network
GPU	Graphics processing unit
IR	Infrared
IS	Inception Score
KL	Kullback Leibler
MMD	Maximum Mean Discrepancy
MT-LB	Mnogotselevoy Tyagach Legky Bronirovanny or Multi- Purpose Towing Vehicle Light Armoured
RGB	Red green blue
RKHS	Reproducing Kernel Hilbert Space
KID	Kernel Inception Distance

1

Introduction

1.1 Motivation

Recent developments have shown that deep neural networks (DNNs) successfully can be used to automatically detect and classify objects in images [11, 24, 25, 30]. This has become very useful within many military and security applications [17, 28, 41]. However, the training of DNNs requires large quantities of annotated data, which is both time consuming and expensive to produce. Within military applications it is at times difficult or even impossible to retrieve due to e.g. a shortage of military vehicles. One way to overcome this is to simulate training data. However, it has been challenging to produce simulations that are sufficiently realistic to not result in a significant performance drop when the learned model is applied to real data [14, 39]. Therefore it is of considerable interest to study different ways to refine simulated training data using image-to-image translation.

1.2 Purpose

The purpose of this study is to investigate possibilities to do image-to-image translation to produce synthetic thermal IR-images that are closer to the domain of authentic thermal IR-images using a cycle-consistent generative adversarial network (GAN). The domain translation is performed either using synthetic IR-images alone, or by also including corresponding semantically annotated data and/or pixelwise depth data. A cycle-consistent GAN includes two neural network models referred to as generators. The goal of the first generator is to learn a mapping from the synthetic domain to the authentic and the other one to learn a mapping for the other way around. If only IR data is used during training, the generator architectures can be the same. If additional semantic and/or depth data is added to the synthetic samples, the generator models will have to be modi-

fied. The generator mapping from the synthetic to the authentic domain has to be able to accept IR-images, as well as their corresponding semantically annotated and/or monocular depth images. This should now map into authentic IR-images. The goal of the generator mapping from the authentic to the synthetic domain is instead to learn a mapping from authentic IR-images to synthetic IR-images as well as their corresponding semantically annotated and/or a monocular depth images. The actual model of interest after training is the generator model which can map from the domain of synthetic to that of authentic data. To evaluate the performance of this generator model, a novel measure based on the Frechét Inception Distance, also known as Wasserstein-2 distance [13], which has been developed to be used specifically for IR-images is proposed. Further, the measure is evaluated to confirm that it works as intended.

1.3 Problem Formulation

Let the goal with a cycle-consistent GAN be to train a neural network model which has learnt a mapping from the domain of synthetic data samples to that of authentic. The question that the study aims to answer is the following: Compared to only including IR-data during training, will the translation of synthetic data samples into the domain of authentic data samples be improved by including corresponding semantic and/or pixelwise depth information to each synthetic data sample?

1.4 Limitations

The largest limitations concern the data. All simulated and authentic images used in the study are based on footage or simulated aerial footage of a military training ground. The training ground includes forests, fields, a few buildings, as well as civilian and military vehicles. The data domain was chosen partly as a consequence of what was available and partly since it is similar to the domain that is of interest to FOI in possible future applications.

Another limitation regards the training and parameter tuning. Due to shortage of time and computational resources, it has not been possible to do any major parameter tuning and the training was limited. Further, only one method for measuring the image-to-image translation results was developed.

2

Related Work

Work within five different domains are considered. These include image-to-image translation in section 2.1, generative adversarial networks (GANs) in section 2.2 and different approaches to evaluate GANs in section 2.3. When adding additional information to the synthetic data when training, i.e. a pixelwise depth map and/or semantic information, the model has to be modified. It results in the generator mapping from the authentic domain to the synthetic domain, having to learn how to map also to a depth map image and/or a semantically segmented image. This is since no depth or semantic data is included in the authentic data and the authentic images mapped to the synthetic domain are used to calculate the loss values which are used to update the models. For this reason, monocular image depth estimation and semantic image segmentation will be considered in section 2.4 and section 2.5.

2.1 Image-to-image translation

Image-to-image translation is mapping of an image from one domain to another. Lately, research within this topic has reached astonishing results [15, 16, 42, 43]. One way to perform image-to-image translation is to use feature-level adaptation methods. Translation of this kind typically involves minimising some measure of distance between the source and the target feature distributions, e.g. as in maximum mean discrepancy [27, 36], adversarial discriminator accuracy [35] or correlation distance [40]. These methods have successfully been used for semi-supervised or unsupervised domain adaptation tasks [27, 36]. The weakness of these methods is that they do not enforce semantic consistency. Further, alignment at a higher level of a deep representation may not create alignment at a lower level of representation, which may be crucial for the final mapping.

Generative pixel-level adaptation models instead perform distribution align-

ment in the pixel space instead of the feature space. This e.g. includes the methods *pix2pix* [16], *cycleGAN* [42], *bicycleGAN* [43] and *MUNIT* [15]. Both the *pix2pix* and the *bicycleGAN* approach require paired training examples, while *cycleGAN* and *MUNIT* do not [15, 16, 42, 43]. According to [16], the *pix2pix* framework is favourable in many image-to-image translation tasks. Especially, if the outputs are highly structured and graphical.

2.2 Generative Adversarial Networks

A generative adversarial network (GANs) is when a generative model is estimated by simultaneously training one generative model, which captures the data distribution, and a discriminative model that aims at distinguishing the distribution of the training data from that of the samples that are being forged by the generator [12]. This type of models have achieved impressive results within image generation [6, 29], representation learning [29, 32] and image-to-image translation [16, 42, 43]. [42] have presented a GAN referred to as *cycleGAN*, which can be trained without paired training data. The *cycleGAN* model includes two generators, G and F , as well as two discriminators D_X and D_Y (fig. 2.1). The task of generator G is to learn to map from domain X to Y , while generator F does the opposite. Discriminator D_Y is set to learn to distinguish between authentic and forged domain Y images, while D_X is set to learn to differentiate authentic and forged images of domain X . *Adversarial losses* are used to match the distribution of generated images to the distribution of the target domain and *cycle-consistency losses* are used to prevent the mappings G and F from contradicting one another [42].

Figure 2.1: The two mapping functions $G : X \rightarrow Y$ and $F : Y \rightarrow X$ and their respective discriminators D_X and D_Y . Figure adapted from [42].

According to [10], circularity constraints such as the cycle-consistency loss used by [42], are not enough to explain the recent success of unsupervised image-to-image translation. Instead, the key to success are architectures that are complex enough to replicate the target distribution based on inputs from the source domain, yet not too complex. According to [10], the target mapping is best ap-

proximated by the lowest-complexity mapping that has a low discrepancy between the mapped samples and the target distribution. In order to find the lowest-complexity mapping, the network cannot be too complex, since it might become distracted by other alignment solutions.

2.3 Evaluation of GANs

The performance of a GAN can be measured using subjective human evaluation of the images produced by a GAN generator. Several other methods that more objectively evaluate the performance of a GAN have been proposed [2, 13, 32]. A common way to evaluate is to measure differences in the statistical distribution between two datasets or representations of two datasets. Typically, one of the datasets consists of images from the target domain and the other of images from the source domain or images from the source domain processed through a GAN generator. Ways to measure the difference between distributions include the Inception Score (IS) [32], the Fréchet Inception Distance (FID) [13] and Maximum Mean Discrepancy (MMD) [2, 36].

IS, first proposed in [32], uses the image classification model called Inception [37], pretrained on ImageNet [5]. The IS is calculated as the Kullback–Leibler divergence between the marginal and the label distributions of images in a dataset processed through the pretrained Inception model. The higher the score the better since a higher score implies a larger difference between these two distributions, i.e. the images are varied and contain meaningful objects. An issue with this method is that if the objects depicted in the images are not among the classes in the Inception model trained on ImageNet, or not all classes of the Inception model are supposed to be represented among the generated images, the score might be low despite the images being well generated.

The Fréchet Inception Distance, also known as Wasserstein-2 distance, was introduced by [13]. It processes all images from two datasets being compared through an Inception-V3 model [37] pretrained on images from ImageNet [5]. The activations from its pool3 layer are retrieved for all images. For each of the datasets, these activation values are approximated as multivariate Gaussian distributions and the Fréchet Distance between these multivariate Gaussians is then calculated. This provides a measure of similarity between the two distributions, where the score is zero if the datasets are identical.

The Maximum Mean Discrepancy calculates distances between two dataset distributions by calculating the distances between mean embeddings of distribution features in a reproducing kernel Hilbert space (RKHS). There are many possible kernel mappings to the RKHS. One is the automatic relevance determination (ARD) kernel over the output dimension as proposed in [36] and one is the *Kernel Inception Distance* (KID) as proposed in [2]. As showed by [21], the FID and KID correlates very strongly.

The goal of refinement of synthetic data is to be able to use the data to train a DNN for object detection. Therefore an alternative approach to measure the improvement of generated data is to train an object detector using the generated

images and evaluate how well the trained model performs on authentic data. This way of measuring the improved training data is a more direct way to measure the performance of the image enhancement. However, it can be more time consuming than using a statistical measure. An advantage with a statistical measure is also that it enables early stopping of the GAN during training.

2.4 Monocular Image Depth Estimation

To be able to use information of depth in each image pixel of the synthetic IR images when training the GAN, the generator translating from the authentic domain to the synthetic has to learn how to map a monocular image to a depth map. This problem is fundamental in computer vision with many applications in 3D-reconstruction, scene understanding and robotics. Thus, several successful attempts to map monocular images to depth maps have been done already [7–9, 22]. The authors of [8] address this issue by using two deep network stacks, where one makes a coarse global prediction based on the whole image and the second refines this prediction locally. They also use a scale-invariant mean squared error in log space to measure the relationships between points in the scene, irrespective of the absolute global scale. Another common practice to calculate the depth loss is to simply use the $\mathcal{L}_1$ norm [3, 19].

2.5 Semantic Image Segmentation

When including semantic information as part of the synthetic data, the GAN generator mapping from authentic images to synthetic will have to learn how to map a thermal IR-image to a semantically segmented image. Convolutional neural networks have proved successful at performing this kind of task. The authors of [26] adapts commonly used classification networks (AlexNet [20], VGG net [34] and GoogLeNet [38]) to the task of segmentation by transforming them into fully convolutional networks and fine-tuning them. They use an encoder-decoder model, where images are downsampled using the base model and then upsampled to original size. By combining semantic information of a deep coarse layer with a shallow fine layer using skip connections, their models produce segmentations that are both accurate and detailed.

A reoccurring problem within semantic segmentation is how to compensate for unbalanced class representations in the training data. One common way is to use *medium frequency balancing*, where each class is weighted with the median class frequency multiplied with the inverse class frequency in the loss function [1, 7]. If the classes used during training are very unbalanced, this can help the model to learn mappings of less frequently seen classes.

3

Method

First, the data used in the study is described in section 3.1. Next, the model implementation is presented in section 3.2 and the objective in section 3.3. Finally, the evaluation method is explained in section 3.4 and the hardware in section 3.5.

3.1 The Data

One dataset with authentic thermal IR-images and one with synthetic thermal IR-images consisting of 10,000 samples each were used. All images are aerial footage or simulated footage based on the military training ground Kvarn [23] (fig. 3.1). Thus the scenes include scenery of forests, fields, a few buildings, as well as civilian and military vehicles. The depression angles range from 10 to 70 degrees and all images are generated such that the vehicles are between 15-40 pixels in width and height. Some of the real images are rotated such that the y-axis is not parallel to the vertical direction, or include motion blur. The IR-images' pixel values are apparent temperature in Kelvin and all data is provided by FOI.

The vehicles in the authentic dataset are sometimes heavily occluded by vegetation [23]. To make the authentic and simulated images as similar as possible, the simulated images also include partly occluded vehicles. Further, some of the vehicles that appear in the real data are not part of the simulated training data. Only vehicles T72, MT-LB and Volvo V70 are included among the synthetic images.

For each synthetic IR-image, there are associated pixelwise depth data and semantic annotations of the complete image represented with one-hot encodings. These were added as extra channels to the synthetic samples if used during training.

The semantic images are originally coloured in a way that makes the class limits ambiguous, which is why they are mapped to distinct classes before being

Figure 3.1: Satellite image of the area from where the authentic training data is from and that the synthetic training data is based on.

Table 3.1: Material classes and their colour representations in the semantically segmented images.

Layer	Material
Gravel and stones	white
Grass surfaces	bright green
Dirt roads	grey
Undervegetation	dark green
Soil areas	bright brown
Asphalt	dark grey
Lakes and shores	blue
Buildings	dark red
Trees	green
Objects	cerise
Ignore	black

used for training. An ignore mask is applied to the pixels in the images where the pixel class is too ill-defined. Examples of the training data can be seen in fig. 3.3 and fig. 3.4. Further, all semantic classes are listed in table 3.1.

The authentic thermal IR images are retrieved from video image footage from an IR sensor hanging from a helicopter, whereas the synthetic data is randomly generated. The camera positions and the corresponding mappings of each camera's optical axis onto an approximated ground plane for all authentic and synthetic images used in the study are as seen in fig. 3.2. The training and test data is split in a way that both training and test data is rather varied. Low similarity between training and test data is ensured by choosing images in a way that the distance between each projection of a camera's optical axis onto the approximated ground plane corresponding to an image from the training data set is always at least 50 meters from one projection corresponding to an image from the test data set. This is also illustrated in fig. 3.2.

3.2 Implementation

First, the network architecture is explained in section 3.2.1, secondly the training details presented in section 3.2.2.

3.2.1 Network Architecture

Since paired training data is unavailable and GANs with cycle-consistency do not require it [42], a cycle-consistent GAN based on a modified version of [42] is used to train a generator model that can be used to make the simulated data more similar to the authentic data. As explained in 2.2, a cycle-consistent GAN consists of two different generators. In this case one generator is supposed to learn a mapping from the domain of synthetic data, X to that of authentic, Y , (gener-

Figure 3.2: How the train and test data is split. Circles represent camera positions and crosses the mappings of each camera’s optical axis onto an approximated ground plane. Darker colours represents training and brighter testing. Axes values are given in meters.

ator G) and one which is supposed to learn a mapping of the other way around (generator F). It also consists of two discriminators, D_Y and D_X . When only IR data is used during training, the generator and discriminator model architectures can have the same architecture, and so can the two discriminators. When semantic and/or depth data is added to the synthetic domain, generator G will map synthetic IR-images, their corresponding semantically annotated images and/or monocular depth images to authentic IR-images. Generator F will instead learn how to predict synthetic IR-images, their corresponding semantically segmented images and/or monocular depth images based on authentic IR-images. Consequentially, discriminator D_X must be able to accept more than one input channel. Thus, four different cycle-consistent GAN models has to be created, hereon referred to as *ir*, *irsem*, *irdepth* and *irsemdepth* depending on what information was included during training. *ir* refers to IR, *sem* to semantic and *depth* to depth data. Next, the *ir* model is explained. This is followed by explanations of how the *ir* model is transformed into the other models to adapt itself to included semantic and/or depth data during training.

The generators in the *ir* model [42] are encoder-decoder models based on a modified version of ResNet. To avoid a too complex model as suggested by [10] and to create a more parameter efficient model for faster performance, a different generator model is used. This model consists of a Mobilenet V2 [33] encoder part as encoder and a U-net decoder part as decoder. Inspired by U-Net [31] and other image-to-image translation algorithms [16], some of the encoder and decoder layers are connected by skips to keep the image structure. Further, the output layer uses a linear activation function, since the output is unbounded. Discriminators D_Y and D_X use PatchGAN, same as in [42]. Unlike [42], the latest generated images are used each time, instead of a buffer of the 50 latest generated images. Below follows a description of how the generators of the *ir* model are

Figure 3.3: Example images from the authentic thermal IR dataset.

Figure 3.4: Examples of images from the synthetic dataset. The left column shows synthetic IR images and the other two their corresponding semantically segmented images and the pixelwise depth map.

modified to be able to train when corresponding semantic and/or depth data is added to the each synthetic data sample.

When the synthetic IR-images are accompanied by their corresponding semantically annotated images and/or monocular depth images during training, each synthetic data sample is fed into generator G as a concatenation along the channels axis. Generator G , which learns to map from the synthetic to the authentic domain, of the *irsem*, *irdepth* and *irsemdepht* models, is adapted to the task by being made into accepting more than one input channel. This enables the additional depth- and/or semantic information to affect the resulting translated images.

To be able to calculate the loss values as explained in section 3.3, generator F must learn to map from authentic thermal IR-images to synthetic thermal IR-images, pixelwise depth- and/or a semantically segmented images, depending on what is included during training. To enable this mapping while adding as few extra parameters to the model as possible, the encoder part of generator F is kept as is for *irsem*, *irdepth* and *irsemdepth*. Instead, the output of the encoder is fed into one decoder which predicts synthetic IR-images, one which predicts semantically annotated images and/or one which predicts monocular depth images. The last layers of the decoder parts that are used to predict either thermal IR data or pixelwise depth values uses a linear activation function since they are unbounded. The last layer of the decoder that predicts semantically segmented images instead uses a softmax activation since it predicts discrete classes. Considering the additional decoder parts added when training with the supplementary data, it was especially important to reduce the number of parameters. This enables creation of a model which is not too large. Lastly, discriminator D_Y is kept as in the *ir* model, while D_X is made into being able to accept more than one input channel. Each sample input into D_X therefore consists of one IR-image, one semantically segmented image and/or one pixelwise depth map.

The encoder-decoder model with a MobileNet V2 [33] encoder as encoder and a U-Net [31] decoder as decoder, with skip connections between some of the layers and a linear activation function, works well to learn a mapping from IR data to pixelwise depth using paired input-output data. The same model, where the activation function instead is a softmax activation function, performs well to learn a mapping from IR data to semantically labelled data. This knowledge further motivated the use of the same encoder part for prediction of IR- and semantic annotation and/or pixelwise depth. It also suggested using similar decoder parts for prediction of semantic annotations and pixelwise depth as for the IR prediction, with the exception of the activation functions.

To conclude, generator G , which is supposed to learn a mapping from the synthetic (X) to the authentic domain (Y), is based on a MobileNet V2 [33] encoder part and an U-Net [31] decoder part. If semantic and/or depth information is added during training, this data is concatenated and fed to the model. Generator F , which maps the other way around, will always have authentic IR-images as input. When only IR-images are used during training, the two generator architectures will be exactly the same. When additional data is added to the synthetic domain, generator F must predict one IR-image, one semantically annotated im-

Figure 3.5: *Left:* Generator $G: X \rightarrow Y$ and *right:* Generator $F: Y \rightarrow X$, where X is the domain of synthetic and Y the domain of authentic data, $x \in X$, $y \in Y$. ir , sem and $depth$ represents IR, semantic and depth data, and variables with a hat are predicted. The arrows connecting layers from the sides are skip connections.

age and/or one monocular depth image. Therefore, it is modified to consist of one MobileNet V2 [33] encoder part, followed by one U-Net [31] decoder part for each of the things it is supposed to predict. Schematic figures of generator models G and F , where semantic and depth data is included during training, can be seen in fig. 3.5. All model implementations are done in Keras and Tensorflow 2.

3.2.2 Training Details

During training, each dataset is split into a training set of 7000 images and a validation set of 3000 images. All images originally have size 240×320 pixels but are reshaped into a size of 256×320 pixels using bilinear interpolation to adapt them to the generator model. All input IR and depth data is normalised by subtraction of the mean and division of the standard deviation of all pixel values of all images used for training. When training using depth data, minibatch standard deviation is applied to prevent mode collapse, as proposed in [18]. The minibatch standard deviation computes the standard deviation for each feature in each spatial location over a whole batch. Using the average standard deviation, an additional constant feature map is created. As suggested by [18], the feature map is then inserted at the end of the discriminator, more specifically before the final discriminator layer. This gives the discriminator information about how varied the images are in a batch, which can prevent mode collapse. Also, the discriminator for synthetic images is penalised when the prediction for a real image goes beyond 0.9 using label smoothing, to avoid overconfidence. All models are trained once for 700 epochs each.

3.3 Objective

First, the objective during training with IR data only is presented, next, how this objective is modified during training with depth and/or semantic information.

Just as in [42], the GAN model makes use of *adversarial losses*, for matching the distributions of the two domains, and *cycle consistency losses* to make sure mappings of the two generators G and F do not contradict each other. In the same way as [42] does for painting generation from photographs, an identity loss is applied, motivated by the fact that the images should be very similar.

As in [42], an adversarial loss is applied to both generators. Let X be the domain of synthetic images and Y the domain of authentic, where each sample $x \in X$ is a synthetic IR-sample. For the mapping function $G : X \rightarrow Y$ and its discriminator D_Y , the objective function is

$$\mathcal{L}_{\text{GAN}}(G, D_Y, X, Y) = \mathbb{E}_{y \sim p_Y(y)}[\log D_Y(y)] + \mathbb{E}_{x \sim p_X(x)}[\log(1 - D_Y(G(x)))], \quad (3.1)$$

$$\mathcal{L}_{\text{GAN}}(G, D_Y, X, Y) = \mathbb{E}_{y \sim p_Y(y)}[\log D_Y(y)] + \mathbb{E}_{x \sim p_X(x)}[\log(D_Y(G(x)))], \quad (3.2)$$

where G attempts to generate images $G(x)$ that are similar to images from domain Y and D_Y tries to distinguish between real samples y and generated samples $G(x)$. Thus G tries to minimise the objective while D_Y tries to maximise it resulting in $\min_G \max_{D_Y} \mathcal{L}_{\text{GAN}}(G, D_Y, X, Y)$. Similarly, the adversarial function for the mapping function $F : Y \rightarrow X$ and its discriminator D_X is $\min_F \max_{D_X} \mathcal{L}_{\text{GAN}}(F, D_X, Y, X)$. In practice, this loss does not work as shown by [12]. Therefore the non-saturating equivalent is used instead, since it has the same optimum, but provides much stronger gradients earlier in the learning process. This amounts to replacing minimisation of the second term in eq. (3.2) with maximisation of $\log(D_Y(G(X)))$ with respect to G .

The cycle consistency loss is expressed as

$$\mathcal{L}_{\text{cyc}}(G, F) = \mathbb{E}_{x \sim p_X(x)}[\|F(G(x)) - x\|_1] + \mathbb{E}_{y \sim p_Y(y)}[\|G(F(y)) - y\|_1] \quad (3.3)$$

and the identity loss as

$$\mathcal{L}_{\text{identity}}(G, F) = \mathbb{E}_{y \sim p_Y(y)}[\|G(y) - y\|_1] + \mathbb{E}_{x \sim p_X(x)}[\|F(x) - x\|_1]. \quad (3.4)$$

The full objective loss is

$$\begin{aligned} \mathcal{L}(G, F, D_x, D_y) = & \mathcal{L}_{\text{GAN}}(G, D_Y, X, Y) + \mathcal{L}_{\text{GAN}}(F, D_X, Y, X) \\ & + \lambda \mathcal{L}_{\text{cyc}}(G, F) + \gamma \mathcal{L}_{\text{identity}}(G, F), \end{aligned} \quad (3.5)$$

where $\lambda = 10$ and $\gamma = 50$ in during all experiments.

Now let each sample $x \in X$ also include a semantically annotated image and/or a pixelwise depth map. When extending each synthetic sample this way,

the original identity loss used to update the model (eq. (3.4)) is now impossible to retrieve since an IR-image with a single channel, $y \in Y$, cannot be processed through a model which now expects several input channels. Thus a new identity loss is used:

$$\mathcal{L}_{\text{identity}}(G) = \mathbb{E}_{x \sim p_X(x)} [\|x_{IR} - G(x)\|_1], \quad (3.6)$$

where x_{IR} is the IR-channel of a synthetic sample x . This new identity loss is used to compensate for the old one and motivated by the fact that the IR part of the X domain and the Y domain should still be very similar.

To update the cycle consistency loss, $\mathcal{L}_{\text{cyc}}$, is modified to include the semantic and depth decoder branches' losses:

$$\mathcal{L}_{\text{cyc}} = \|y - G(F(y))\|_1 + \|x - F(G(x))\|_1 + \alpha \mathcal{L}_{\text{semantic}} + \beta \mathcal{L}_{\text{depth}}, \quad (3.7)$$

where Keras' [4] Sparse Categorical Cross entropy is used to calculate $\mathcal{L}_{\text{semantic}}$ and weighted with the inverse frequencies of the corresponding classes in the training dataset. This is to avoid misclassifications of classes that are very scarce in the dataset. The depth loss, $\mathcal{L}_{\text{depth}}$, is the $\mathcal{L}_1$ norm:

$$\mathcal{L}_{\text{depth}} = \mathbb{E}_{x \sim p_X(x)} [\|x_{\text{depth}} - F(G(x))_{\text{depth}}\|_1], \quad (3.8)$$

where x_{depth} is the ground truth depth map and $F(G(x))_{\text{depth}}$ is the depth map predicted by generator F , based on a prediction of an authentic IR-image by generator G . Weight α is empirically set to one for all experiments and β is set to 500 when only IR and depth information is used for training, otherwise one.

3.4 Evaluation

The evaluation of the image refinement is done using a modified version of the Frchet Inception Distance (FID) [13]. The FID is used partly since it has proved to capture the similarity between datasets better than the Inception Score (IS) [13] and partly due to its relatively common usage and easy implementation. To adjust the measurement to IR-images, an encoder-decoder model trained on semantically segmenting IR images is used instead of the Inception model. The encoder part is based on Mobilenet V2 [33] with a decoder based on U-Net [31] with up-sampling layers and several skip connections between the encoder and decoder to keep structure, same as the generator models used in the cyclic GAN model explained above. IR images with corresponding semantically segmented images used to train the model is provided by FOI. To compensate for class representation imbalance in the dataset, the loss values are weighted with the inverse frequencies of the corresponding classes in the training dataset on a pixel level inspired by [7]. Examples of training data and learnt mappings can be seen in fig. 3.6.

To calculate the new distance measure between two datasets, activations from layer 'block_16_project' in the Mobilenet V2 encoder part are retrieved for all images in each dataset. For each image, this results in a tensor with shape $8 \times 10 \times 320$.

Figure 3.6: Examples of IR images and corresponding learned semantic segmentation mappings. Some of the examples include more correctly labelled pixels than the others.

Since the segmentation network is shift-invariant and objects can appear anywhere in the images, each spatial position can be seen as an observation. Thus, there are 80 observations of 320 channels for each image processed through the model. As in [13], these activations are assumed to follow a multidimensional Gaussian distribution. For each set of activations produced by each dataset, both mean and covariance is calculated. Let $\mathcal{N}(\mu_1, C_1)$ and $\mathcal{N}(\mu_2, C_2)$ be the multivariate Gaussian distributions of the 320-dimensional activation vectors for each dataset. The Frechét distance between the authentic and the generated dataset activation distributions, d , is now calculated as

$$d^2 = \|\mu_1 - \mu_2\|^2 + \text{Tr}(C_1 + C_2 - 2(C_1 * C_2)^{\frac{1}{2}}). \quad (3.9)$$

To show that the measure works as intended, it is evaluated by calculating the FD between two sets of 3500 images each, where the first set consists of authentic thermal IR-images and the second of images from the same distribution, but with added distortion. Distortions are either different amounts of Gaussian noise or blur or contamination with IR-images from a different dataset. One of the datasets used for contamination consists of synthetic thermal IR-images. The second dataset consists of the thermal IR-images used to train the model which yields the activations which are used to calculate the Frechét distance measure. When the compared datasets measured are identical, the measure is zero. When the compared datasets are not identical, but drawn from the same distribution, the measurement is 0.316. The low, close to zero value here is expected since the distributions are not exactly the same, only equivalent. The measure increases monotonically with increasing Gaussian noise and blur. When the dataset is contaminated, the distance is largest when half of the images of the second dataset are authentic and the other half is synthetic and smaller the more of a single dataset is included (fig. 3.7). Since the dominant part of the sum in eq. (3.9) is the covariance term, this is to be expected. Still, there is a distinct difference between when the compared datasets are from the same distribution and when they are not. The FD value using 100% synthetic images is 4.84 and this is used as a baseline.

3.5 Hardware

All tests are run on a Tyan server with two Intel Xeon Gold 6154 3 GHz CPUs with 786 GB RAM and eight Nvidia Quadro RTX 6000 GPUs and Linux Ubuntu 18.04 LTS installed.

Figure 3.7: FD measure evaluated using **left:** Gaussian noise, **middle:** Gaussian blur and **right:** contamination with a second dataset. The datasets utilized for contamination is either the synthetic thermal IR-images or an extra authentic thermal IR dataset. The extra dataset consists of the thermal IR-images used for training the model which yields the activations used to calculate the Frechét distance. An average value of ten evaluations is used for each test.

4

Results

First, the resulting FD values for the different experiments are presented and secondly examples of translated images.

The evaluation using the Frechét Distance (FD) shows that all trained models have learned to generate better images than the baseline (fig. 4.1). This implies that all trained models manage to move the statistical distribution of the synthetic data closer to that of the authentic domain. Based on the FD scores, training using only IR data translates images to the authentic domain the best (fig. 4.1).

A set of synthetic and authentic images can be seen in fig. 4.2 and fig. 4.3 respectively. Using the model with the best FD score, the synthetic images have been translated to the authentic domain using generator G and then reconstructed using generator F . The authentic IR-images have been translated to the synthetic domain using generator F . The resulting images are shown and discussed below. Since vehicles are of considerable interest for future applications, only images with vehicles are included among the examples.

Except for more blur, it is difficult to see any major changes to the images generated using the model trained only with IR data only (fig. 4.4). The authentic images translated to the synthetic domain by generator F (fig. 4.5) have minor temperature differences and smoothed trees and fields. Except for that, they look identical to the original images.

The images generated with the model trained using IR and semantic data also appear to be blurred compared to the original IR-images. Further, they show distortions at some places belonging to the undergrowth class (fig. 4.6). There are also some pixels in the upper right corners which show major distortions. Reconstruction of translated synthetic images works well for the IR domain, while the semantic reconstruction struggles, mostly with the building class and confuses the clay class with the dirt roads class. Still, it maps vehicles very well. The generator F mapping from authentic images to synthetic is more challenging (fig. 4.7).

Figure 4.1: FD-values for each generator G model. *ir*, *sem* and *depth* refer to IR, semantic and depth information and specify what was used during training of each model. The dashed line represents the the baseline, which is determined by evaluating the FD for 100% synthetic images.

Figure 4.2: Examples of synthetic thermal IR-images and corresponding semantically segmented images and pixelwise depth maps.

Figure 4.3: Examples of authentic thermal IR-images.

The IR-images show a grid pattern and the semantic images mostly manage to correctly map vehicles, while it struggles with other classes, mapping most of the semantic images to grass.

Using IR and depth data during training, the synthetic data translated to authentic thermal IR-images show blur and some traces of a grid like pattern (fig. 4.8). Reconstruction of the same images indicate that the model has learnt to map the pixelwise depth fairly well, while the reconstructed IR-images are still blurred and show a grid like pattern (fig. 4.8). The translation of authentic images (fig. 4.3) to synthetic (fig. 4.9), show a minor grid like pattern in the translated IR-images. The model has learnt that the lower part of the images is usually closer than the upper part. However, all depth maps are quite similar and it does not succeed at correctly mapping trees and other objects in the images.

The IR-images generated with the model trained using all available data performs second best according to the FD values (fig. 4.1). Synthetic images translated to the authentic domain using this model are as seen in fig. 4.10. The translated images show distortions where the undergrowth class is, but barely any hallucinatory patterns. The reconstructed semantic images are more similar to the original semantic images than the reconstructed images using the model trained only using IR and semantic data. The reconstructed depth maps show contours of trees, where some objects show a grid pattern. The translation of authentic images to synthetic using generator F of this model performs equally to the other models (fig. 4.11). The translated authentic IR-images are very similar to the original authentic IR-images (fig. 4.3) and the depth maps show that the model has learnt that the lower parts of the images are usually closer to the camera than the upper parts, while it struggles with objects. Most of the semantic images are mapped to grass, while the vehicles are often mapped correctly and it struggles with all other classes.

To conclude, the model performing best according to the FD score is training

Figure 4.4: The first two image rows are the IR-images from fig. 4.2, mapped to the authentic domain using generator G . The two following rows are the same images mapped back to the synthetic domain using generator F . The data used to train the model is limited to IR-images.

Figure 4.5: The images from fig. 4.3 mapped to the synthetic domain using generator F . The model is trained on data limited to IR-images.

using only IR data. Second best is the model using IR, semantic and depth data. Common to all translated synthetic images is that the models have learned to remove high frequencies in the images. Specifically, there are no major changes to vehicle appearance visible in generated images for any of the models, except for blurring of lines. Further, the images generated with the model trained using IR-images only, include less grid-pattern distortions than the other images.

Figure 4.6: The images in the first two rows are images from fig. 4.2, mapped to the authentic domain using generator G . The following rows contains reconstructions of the same images using generator F . The models are trained using both IR and semantic information.

Figure 4.7: The images from fig. 4.3, mapped to the synthetic domain using generator F trained on IR and semantic data.

Figure 4.8: The images in the first two rows are images from fig. 4.2, mapped to the authentic domain using generator G. The following rows contains reconstructions of the same images using generator F. The models are trained using both IR and depth information.

Figure 4.9: The images from fig. 4.3, mapped to the synthetic domain using generator F trained on IR and depth data.

(a) The images in the first two rows are images from fig. 4.2, mapped to the authentic domain using generator G . The following two rows contains IR reconstructions of the same images using generator F . The models are trained with IR, semantic and depth data.

Figure 4.10: Translated and reconstructed images from fig. 4.2 using the model trained with IR, semantic and depth data.

(b) Semantic and depth map reconstructions of the images in the first two rows in (a) using generator F . The models are trained with IR, semantic and depth data

Figure 4.10: Translated and reconstructed images from fig. 4.2 using the model trained with IR, semantic and depth data.

Figure 4.11: The images from fig. 4.3, mapped to the synthetic domain using generator F trained on IR, semantic and depth data.

5

Discussion

First, the results and the method are discussed in section 5.1 and section 5.2. Further, possible future work is considered in section 5.3.

5.1 Results

As stated in 4, training with only IR data obtained the best FD score. Since the trained models might differ between runs depending on the randomly initiated weights, an average best FD over several runs should be used to get a more trustworthy result. However, this is very computationally demanding.

5.2 Method

The distortions caused by the *undergrowth* class and the challenge of distinguishing between the two classes *dirt roads* and *clay* in the resulting images, imply that it is of interest to either create better synthetic data where the undergrowth class only exists under trees or all undergrowth should instead be grass. Furthermore it is probably better if the dirt roads and clay class are the same class in future experiments.

Instance normalisation is used in all network layers, in both the generators and discriminators, except for the generator encoder part consisting of MobileNet v2 [33]. In a recent study, [21] shows that spectral normalisation can greatly improve GANs. It would therefore be of interest to use this for all layers in future experiments.

How much of the IR-images produced by generator G are affected by the middle layers of the model or whether the model has simply learnt to use the skip connection between the first and the last layers of the model is unclear. To investi-

gate it further, it would be of interest to study how much of the signal is actually passed from the very next last layer into the final layer of the model.

It could also be of interest to test more different combinations of learning rates and different loss weights for the semantic and depth losses, to optimise the system. Possibly, the result of the models trained with semantic and/or depth data can be improved using gradually increasing or decreasing semantic and depth weights.

The models where semantic and/or depth information is included in the training hold many more parameters than training with only IR data. Possibly, the reason why the model trained using only IR data performs the best, is since it is the only model that has had time to converge within 700 epochs. The fact that the images generated using the model trained using only IR data has less grid pattern distortions than the other resulting images, strengthen this hypothesis. Before concluding that training using IR-images only is the best way to get a good FD score, the models should therefore be trained for more than 700 epochs.

To conclude, the results show that it is possible to use GANs to move the distribution of synthetic training data closer to authentic training data. This opens up for quicker and cheaper generation of training data, which is relevant in both military and civilian applications.

5.3 Future Work

Except for the above mentioned method improvements, it is of great interest to evaluate the generated data and the FD measure. How well does an object detector trained on translated synthetic data perform on authentic data? Finally, how well does a good FD measure correlate with a good object detector?

6

Conclusion

In this work, translations of synthetic thermal IR-images to the domain of authentic thermal IR-images using GANs have been investigated. Experiments have been done by adding semantic and/or pixelwise depth information to the synthetic images and the results evaluated with a measure, FD, based on the Frechét Inception Distance (FID). The study concludes that training of the model using IR images alone moved the generated images closest to the domain of authentic thermal IR images since it earned the best FD score. However, this might be due to the fact that the model trained using only IR-images is the only model that had time to converge within 700 epochs. To conclude which model can get the best FD score, it is therefore necessary to train all models for more epochs than 700. It remains for future work to try different training settings to optimise the system and to use the translated synthetic data for training of an object detector. It also remains for the future to thoroughly evaluate the correlation of a good FD score with translated data that works well as training data for a DNN. This study gives a valuable insight into how synthetic data best can be improved to better work as training data for a DNN, which is a vital step to obtain training data quickly and at low cost.

Bibliography

- [1] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(12):2481–2495, 2017.
- [2] Mikołaj Bińkowski, Dougal J. Sutherland, Michael Arbel, and Arthur Gretton. Demystifying MMD GANs. In *International Conference on Learning Representations*, 2018. URL <https://openreview.net/forum?id=r1lUOzWCW>.
- [3] Yuhua Chen, Wen Li, Xiaoran Chen, and Luc Van Gool. Learning Semantic Segmentation From Synthetic Data: A Geometrically Guided Input-Output Adaptation Approach. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.
- [4] François Chollet et al. Keras. <https://github.com/fchollet/keras>, 2015.
- [5] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.
- [6] Emily L Denton, Soumith Chintala, Arthur Szlam, and Rob Fergus. Deep Generative Image Models using a Laplacian Pyramid of Adversarial Networks. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 1486–1494. Curran Associates, Inc., 2015. URL <http://papers.nips.cc/paper/5773-deep-generative-image-models-using-a-laplacian-pyramid-of-adversarial-networks.pdf>.
- [7] David Eigen and Rob Fergus. Predicting Depth, Surface Normals and Semantic Labels With a Common Multi-Scale Convolutional Architecture. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.
- [8] David Eigen, Christian Puhrsch, and Rob Fergus. Depth Map Prediction from a Single Image using a Multi-Scale Deep Network. In Z. Ghahramani,

- M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2366–2374. Curran Associates, Inc., 2014. URL <http://papers.nips.cc/paper/5539-depth-map-prediction-from-a-single-image-using-a-multi-scale-deep-network.pdf>.
- [9] Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, and Dacheng Tao. Deep Ordinal Regression Network for Monocular Depth Estimation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.
- [10] Tomer Galanti, Lior Wolf, and Sagie Benaim. The Role of Minimal Complexity Functions in Unsupervised Learning of Semantic Mappings. In *International Conference on Learning Representations*, 2018. URL <https://openreview.net/forum?id=H1VjBebR->.
- [11] Ross Girshick. Fast R-CNN. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2015.
- [12] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc., 2014. URL <http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf>.
- [13] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 6626–6637. Curran Associates, Inc., 2017. URL <http://papers.nips.cc/paper/7240-gans-trained-by-a-two-time-scale-update-rule-converge-to-a-local-nash-equilibrium.pdf>.
- [14] Stefan Hinterstoisser, Olivier Pauly, Hauke Heibel, Marek Martina, and Martin Bokeloh. An Annotation Saved is an Annotation Earned: Using Fully Synthetic Training for Object Detection. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pages 2787–2796, 2019.
- [15] Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal Unsupervised Image-to-image Translation. In *The European Conference on Computer Vision (ECCV)*, September 2018.
- [16] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-To-Image Translation With Conditional Adversarial Networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

- [17] Farrukh Kamran, Muhammad Shahzad, and Faisal Shafait. Automated Military Vehicle Detection from Low-Altitude Aerial Images. In *2018 Digital Image Computing: Techniques and Applications (DICTA)*, pages 1–8, 2018.
- [18] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive Growing of GANs for Improved Quality, Stability, and Variation. In *International Conference on Learning Representations*, 2018. URL <https://openreview.net/forum?id=Hk99zCeAb>.
- [19] Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-Task Learning Using Uncertainty to Weigh Losses for Scene Geometry and Semantics. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.
- [20] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012. URL <http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf>.
- [21] Karol Kurach, Mario Lučić, Xiaohua Zhai, Marcin Michalski, and Sylvain Gelly. A Large-Scale Study on Regularization and Normalization in GANs. In *International Conference on Machine Learning*, 2019. URL <https://arxiv.org/abs/1807.04720>.
- [22] Lubor Ladicky, Jianbo Shi, and Marc Pollefeys. Pulling Things out of Perspective. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.
- [23] Lage Svensson and Magnus Elmquist and Tomas Chevalier. Semark Fältförsök 2004. *FOI Memo 1009*, 2004.
- [24] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar. Focal Loss for Dense Object Detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.
- [25] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. SSD: Single Shot MultiBox Detector. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision – ECCV 2016*, pages 21–37, Cham, 2016. Springer International Publishing. ISBN 978-3-319-46448-0.
- [26] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully Convolutional Networks for Semantic Segmentation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.
- [27] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan. Learning transferable features with deep adaptation networks. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on*

- Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 97–105, Lille, France, 07–09 Jul 2015. PMLR. URL <http://proceedings.mlr.press/v37/long15.html>.
- [28] Muhammad A. Ma’sum, M. K. Arrofi, Grafika Jati, Futuhal Arifin, Muhammad N. Kurniawan, Petrus Mursanto, and Wisnu Jatmiko. Simulation of intelligent Unmanned Aerial Vehicle (UAV) For military surveillance. In *2013 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, pages 161–166, 2013.
 - [29] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. In *International Conference on Learning Representations (ICLR)*, November 2016.
 - [30] Joseph Redmon and Ali Farhadi. YOLO9000: Better, Faster, Stronger. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.
 - [31] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Cham, 2015. Springer International Publishing. ISBN 978-3-319-24574-4.
 - [32] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, Xi Chen, and Xi Chen. Improved Techniques for Training GANs. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 2234–2242. Curran Associates, Inc., 2016. URL <http://papers.nips.cc/paper/6125-improved-techniques-for-training-gans.pdf>.
 - [33] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4510–4520, 2018.
 - [34] Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *International Conference on Learning Representations (ICLR)*, 2014.
 - [35] Baochen Sun and Kate Saenko. Deep CORAL: Correlation Alignment for Deep Domain Adaptation. In Gang Hua and Hervé Jégou, editors, *Computer Vision – ECCV 2016 Workshops*, pages 443–450, Cham, 2016. Springer International Publishing. ISBN 978-3-319-49409-8.
 - [36] Dougal J. Sutherland, Hsiao-Yu Fish Tung, Heiko Strathmann, Soumyajit De, Aaditya Ramdas, Alexander J. Smola, and Arthur Gretton. Generative Models and Model Criticism via Optimized Maximum Mean Discrepancy. In

- International Conference on Learning Representations (ICLR)*, 2017. URL <https://openreview.net/pdf?id=HJWHIKqgl>.
- [37] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going Deeper With Convolutions. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.
 - [38] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–9, 2015.
 - [39] Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 23–30, 2017.
 - [40] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial Discriminative Domain Adaptation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.
 - [41] Guozhao Zeng, Rui Song, Xiao Hu, Yueyue Chen, and Xiaotian Zhou. Applying Convolutional Neural Network for Military Object Detection on Embedded Platform. In Weixia Xu, Liquan Xiao, Jinwen Li, and Zhenzhen Zhu, editors, *Computer Engineering and Technology*, pages 131–141, Singapore, 2019. Springer Singapore. ISBN 978-981-13-5919-4.
 - [42] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired Image-To-Image Translation Using Cycle-Consistent Adversarial Networks. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.
 - [43] Jun-Yan Zhu, Richard Zhang, Deepak Pathak, Trevor Darrell, Alexei A Efros, Oliver Wang, and Eli Shechtman. Toward Multimodal Image-to-Image Translation. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 465–476. Curran Associates, Inc., 2017. URL <http://papers.nips.cc/paper/6650-toward-multimodal-image-to-image-translation.pdf>.