remote sensing

MDPI

Article

Do Game Data Generalize Well for Remote Sensing Image Segmentation?

Zhengxia Zou* [1], Tianyang Shi* [2,3,4], Wenyuan Li [2,3,4], Zhou Zhang [5], and Zhenwei Shi [2,3,4]

[1] Department of Computational Medicine and Bioinformatics, University of Michigan, Ann Arbor, MI 48109, USA;
[2] Image Processing Center, School of Astronautics, Beihang University, Beijing 100191, China;
[3] Beijing Key Laboratory of Digital Media, Beihang University, Beijing 100191, China
[4] State Key Laboratory of Virtual Reality Technology and Systems, School of Astronautics, Beihang University, Beijing 100191, China
[5] Department of Biological Systems Engineering, University of Wisconsin-Madison, Madison, WI 53706, USA;
* Correspondence: Zhenwei Shi (shizhenwei@buaa.edu.cn)

Received: ; Accepted: ; Published:

Abstract: Despite the recent progress in deep learning and remote sensing image analysis, the adaption of a deep learning model between different sources of remote sensing data still remains a challenge. This paper investigates an interesting question: do synthetic data generalize well for remote sensing image applications? To answer this question, we take the building segmentation as an example by training a deep learning model on the city map of a well-known PC game "Grand Theft Auto V" and then adapting the model to real-world remote sensing images. We propose a Generative Adversarial Networks (GAN) based segmentation framework to improve the adaptability of the segmentation model. Our model consists of a CycleGAN model and a ResNet based segmentation model, where the former one is a well-known image to image translation framework which learns a mapping of the image from the game domain to the remote sensing domain; and the latter learns to predict pixel-wise building masks based on the transformed data. All models in our method can be trained in an end-to-end fashion. The segmentation model can be trained without requiring any additional ground truth reference of the real-world images. Experimental results on a public building segmentation dataset suggest the effectiveness of our adaptation method. Our method outperforms some other state of the art semantic segmentation methods, e.g. Deeplab-v3 and UNet. Another advantage of our method is that by introducing semantic information to the image to image translation framework, the image style conversion can be further improved.

Keywords: Remote sensing; deep learning; video game; domain adaptation; building segmentation.

1. Introduction

Remote sensing has opened a door for people to better understand the earth, changing all walks of our life. Remote sensing technology has very broad applications, including disaster relief, land monitoring, city planning, etc. With the rapid development of imaging sensors, the modality of the remote sensing data is becoming more and more diversified. People now can easily acquire and access to up-to-date remote sensing images from a variety of imaging platforms (e.g., airborne, spaceborne) with a wide spectral range (from multi-spectrum to hyper-spectrum) at multiple spatial resolutions (from centimeters to kilometers).

Recently, the deep learning technology [1,2] have drawn great attention in a variety of research fields. The deep Convolutional Neural Networks (CNN) [3–6] proves to be much more effective than traditional multi-layer perceptron in computer vision and image processing tasks, such as object

detection [7–12], semantic segmenation [13–15], image captioning [16–18], image super-resolution [19–21], etc. A CNN which consist of multiple convolutional and down-sampling layers, learning high-level image abstraction of the data with better discrimination and robustness as opposed to that in traditional methods, where features have to be handcrafted designed. The deep CNNs have also greatly promoted the progress of remote sensing technology [22–26].

Despite its recent success in automatic remote sensing image analysis, the adaption of a deep learning model between different sources of remote sensing data still remains a challenge. On one hand, most of the previous methods of this field are designed and trained based on images of specific resolution or motility. When these methods are applied across different platforms (e.g., the remote sensing images with different modality or image resolution), their performance will be deeply affected. On the other hand, with the fast increase of the deep neural networks' capacity, the training of deep learning models requires a huge amount of data with high-quality annotations, but the manual labeling of these data is time-consuming, expensive and may heavily rely on domain knowledge.

Arguably, improving the adaptability of a model between the different sources of images can be essentially considered as a visual domain adaptation problem [29–31]. The mechanism behind the degradation lies in the none independent and identically distributed data between the training and deployment, i.e., the "domain gap" [29,32] between different sources. Since the training of most of the deep CNN models can be essentially considered as a maximum likelihood estimation process under the assumption of the "independent and identically distribution" [2], once the data distribution has changed after training, the performance can be deeply affected. An important idea for tackling this problem is to learn a mapping/transformation between the two groups of data (e.g., the training data and the testing data) so that they will have, in principle, the same distribution.

In the computer vision field, efforts have been made to generalize a model trained on the rendered images [27,28] (e.g., computer games) to real-world computer vision tasks and have obtained promising results [33–36]. In recent open-world PC games, such as Grand Theft Auto[1], Watch Dogs[2], and Hitman[3], to improve a player's immersion, the game developers feature extensive and highly realistic worlds. The realism of these games is not only in the high fidelity of material appearance and light transport simulation but also in the content of the game worlds: the layout of structures, objects, vehicles and environments [28]. In addition to realism, the game maps are growing explosively and are made more and more sophisticated. For example, in a well-known game "Grand Theft Auto V (GTA-V)" [4], the Los Santos, a fictional city featured in the game's open world, covers an area of over 100km^2 with unprecedented details. Reviewers praised its design and similarity to Los Angeles. Fig. 1 shows a part of its official map and a frame rendered by the image.

In this paper, we investigate an interesting question: do synthetic data generalize well for remote sensing applications? To answer this question, we train a remote sensing image segmentation model on the city map of the well-known video game GTA-V and then adapting it to real-world remote sensing application scenarios. Due to the "domain gap" between the game data and the real-world data, simply applying the models trained on game data may lead to a high generalization error on real-world applications. A general practice to tackle this problem is "neural style transfer" i.e. to transform the game images by using a deep neural network so that the transformed images share the similar style of the real-world ones while keeping their original image contents unchanged [37–40]. The Fully Convolutional Adaptation Network (FCAN) [33] is a representative of this group of the method. The FCAN aims to improve the street-view image segmentation across different domains. By transforming the game data to the style of urban street scenes based on neural style transfer, it narrows the domain gap and improves the segmentation performance. More recently, Generative Adversarial

1 https://www.rockstargames.com/
2 https://www.ubisoft.com/en-us/game/watch-dogs/
3 https://hitman.com/
4 https://www.rockstargames.com/V/

Figure 1. An official map of the PC game GTA-V: the city of Los Santos. (a) The satellite imagery rendered from aerial view. (b) An in-game frame rendered from the "first-person perspective". (c) A part of the game map that is used in our experiment. (d) The legend of the map (in a similar fashion of Google maps). Different from the previous datasets [27,28] that focuses on rendering street-view images from the "first-person perspective" (like (b)), we build our dataset from the "aerial perspective" of the city (c-d).

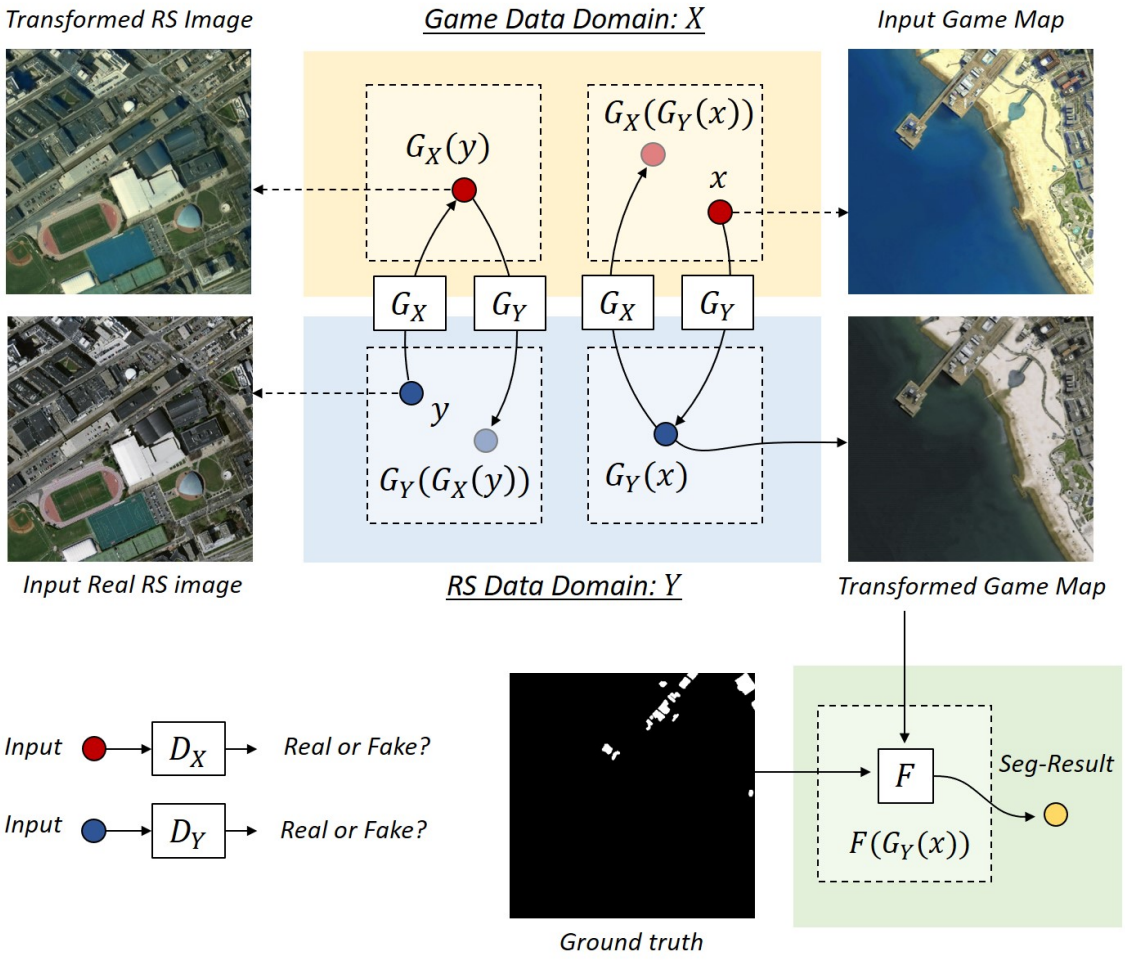

Figure 2. An overview of our method. Our method consists of five networks: G_X, G_Y, D_X, D_Y, and F. The former four networks learn two mappings between the game domain X and the remote sensing domain Y ($G_X : Y \rightarrow X$, $G_Y : X \rightarrow Y$). The last network F learns to predict building masks of the transformed data. In our method, we first transfer the style of a synthetic game map x to a real one $G_Y(x)$ and then train the network F based on the transformed image $G_Y(x)$ (input) and the game map legend (ground truth).

Networks (GAN) [41,42] has greatly promoted the progress of image translation [41,43–45]. Zhu *et al.* proposed a method called CycleGAN [44], which has achieved impressive results in a variety of image to image translation tasks. Owing to the "cycle consistency loss" they introduced, people are now able to obtain realistic transformations between the two domains even without the instruction of paired training data. CycleGAN has then been applied to improve visual domain adaptation tasks [34,46].

In this paper, we choose an important application in remote sensing image analysis, i.e. the building segmentation, as an example by training a deep learning model on the city map of the game GTA-V and then adapting our model to real-world remote sensing images. We build our dataset based on the GTA-V official game maps. Different from any previous methods [33–36] and any previous image segmentation dataset [27,28] that focuses on the game images generated from the "first-person perspective" or from the "street view", our dataset is built from an "aerial view" of the game world, resulting more abundant ground features and spatial relationship of different ground objects.

We further proposed a generative adversarial training based method, called "CycleGAN based Fully Convolutional Networks (CGFCN)", on top of the CycleGAN [44], to improve the adaptability of a deep learning model to different sources of remote sensing data. Our model consists of a CycleGAN model [44] and a deep Fully Convolutional Networks (FCN) [13,47] based segmentation model, where

the former one learns to transforms the style of an image from the "game domain" to the "remote sensing domain" and the latter one learns to predict pixel-wise building masks. The two models can be trained in an end-to-end fashion without requiring any additional ground truth reference of the real-world images. Fig. 2 shows an overview of our method.

Different from the previous methods like FCAN [33] where the image transformation model and the segmentation model are trained separately, our model can be jointly trained in a unified framework, which leads to additional performance gains. Experimental results on Massachusetts Buildings [48], a well-known building segmentation dataset, suggest the effectiveness of our adaptation method. Our method outperforms some other state of the art semantic segmentation methods, e.g. Deeplab-v3 [47] and UNet [49]. In addition, by introducing semantic information to the image to image translation framework, the image style conversion of the CycleGAN can be further improved by using our method.

The contributions of this paper are summarized as follows:

- We investigate an interesting question: do game data generalize well for remote sensing image segmentation? To answer this, we study the domain adaptation ability of a deep learning based segmentation methods by training our model based on the rendered in-game data and then apply it to real-world remote sensing tasks.
- We introduce a synthetic dataset for building segmentation based on the well-known PC game GTA-V. Different from the previous datasets [27,28] that focuses on rendering street-view images from the "first-person perspective", we build our dataset from the "aerial perspective" of the city. To our best knowledge, this is the first synthetic dataset that focuses on aerial view image segmentation tasks. We will make our dataset publicly available.

The rest of this paper is organized as follows. We give a detailed introduction to our method in Section 2. Our experimental datasets and evaluation metrics are introduced in Section 3. The experimental results are given in Section 4, and the conclusions are drawn in Section 5.

2. Methodology

In this section, we will first give a brief review of some related methods, including the vanilla GAN [41] and the CycleGAN [44]. Then, we will introduce the proposed CGFCN and our implementation details.

2.1. Generative Adversarial Networks (GAN)

GAN was originally proposed by A. Goodfellow *et al.* in 2014 [41]. It has then received creasing attentions and achieved impressive results in various computer vision tasks, such as image generation [50–53], image to image translation [44,54,55], object detection [56–58], image super-resolution [21,59,60], etc.

The key to GAN's success is the idea of adversarial training where the two networks, a generator G and a discriminator D, will contest with each other in a minimax two-player game and forces the generated data to be, in principle, indistinguishable from real ones. In this framework, the generator aims to learn a mapping $G(z)$ from a latent noise space $z \in \mathbb{Z}$ to a particular data distribution of interest. The discriminator, on one hand, aims to discriminate between instances from the true data distribution $x \sim p_{data}$ and those generated ones $G(z)$, on the other hand, feeds its output back to G to further make the generated data indistinguishable from real ones. The training of a GAN can be considered as solving the following minimax problem:

$$\begin{aligned}\min_G \max_D V(D,G) = &\mathbb{E}_{x\sim p_{data}(x)}\{\log D(x)\} \\ &+ \mathbb{E}_{z\sim p_z(z)}\{\log(1-D(G(z)))\},\end{aligned} \tag{1}$$

where x and z represent a true data point and an input random noise. The above problem can be well-solved by iteratively updating D and G: i.e., by first fixing G and updating D to maximize $V(D,G)$, and then fixing D and updating G to minimize $V(D,G)$. As the adversarial training progresses, the D will have more powerful discriminative ability and thus the images generated by the G will become more and more realistic. As is suggested by I. Goodfellow *et al.*[41], instead of training G to minimize $\log(1-D(G(\cdot)))$, in practice, many researchers choose to maximize $\log D(G(\cdot))$. This is because in the early stage of learning, $\log(1-D(G(\cdot)))$ tends to saturate. This revision on objective provides much stronger gradients.

2.2. CycleGAN for Image to Image Translation

Suppose X represents a source domain (e.g., the game maps), Y represents a target domain (e.g., the real-world remote sensing images), and $x_i \in X$ and $y_j \in Y$ are their training samples. In a GAN-based image to image translation task [44,54], we aim to learn a mapping G: $X \rightarrow Y$ such that the distribution of images from $G(x_i)$ is indistinguishable from the distribution of Y using an adversarial loss. In this case, the above random noise vector z in the vanilla GAN will be replaced by an input image x. In addition, the generator G and the discriminator D are usually constructed based on deep convolutional networks [51]. Similar to the vanilla GAN [41], the G and D are also trained to compete with each other. Their objective function can be rewritten as follows:

$$\begin{aligned}\min_G \max_D \mathcal{L}(G,D) = &\mathbb{E}_{x\sim p_x(x)}\{\log(1-D(G(x)))\} \\ &+ \mathbb{E}_{y\sim p_y(y)}\{\log D(y)\}\end{aligned} \tag{2}$$

where x and y represent two images from the domain A and domain B. $p_x(x)$ and $p_y(y)$ are their data distributions.

In 2017, Zhu *et al.* proposed CycleGAN [44] for solving the image to image translation problem. The main contribution of the CycleGAN is the introduction of the cycle "consistency loss" in the adversarial training framework. CycleGAN breaks the limits of previous GAN based image translation methods, in which their models need to be trained by pair-wise images between the source and target domains. Since no pair-wise training data is provided, they couple it with an inverse mapping F: $Y \rightarrow X$ and enforce $F(G(X)) \approx X$ (and vice versa).

A CycleGAN consists of four networks: two generative networks G_Y, G_X, and two discriminative networks D_Y, D_X. To transform the style of an image $x_i \in X$ to the domain Y, a straight forward implementation would be training the G_Y to learn a mapping from X to Y so that to fool the D_Y to make it fail to tell which domain they belong to. The objective function for training the G_Y and D_Y can be thus written as follows:

$$\begin{aligned}\mathcal{L}_{X\rightarrow Y}(G_Y,D_Y) = &\mathbb{E}_{y\sim p_y(y)}\{\log D_Y(y)\} \\ &+ \mathbb{E}_{x\sim p_x(x)}\{\log(1-D_Y(G_Y(x)))\},\end{aligned} \tag{3}$$

where $G_Y(x)$ maps the data from domain X to domain Y, and $D_Y(G_Y(x))$ is trained to classify whether the transformed data is real or fake. Similarly, G_X can also be trained to learn to map the data from Y to X and D_X is trained to classify it. The objective function for the training of G_X and D_X can be thus written as $\mathcal{L}_{Y\rightarrow X}(G_X,D_X)$.

However, because the mapping is highly under-constrained, with large enough model capacity, the networks G_X and G_Y can map the same set of input images to any random permutation of images in the target domain if no pair-wise training supervision is provided [44], thus may fail to learn image correspondence between the two domains. To this end, the CycleGAN introduces a cycle consistency

loss that further enforces the transformed image to be mapped back to itself in the original domain: $x \to G_Y(x) \to G_X(G_Y(x)) \approx x$. The cycle consistency loss is defined as follows:

$$\begin{aligned}\mathcal{L}_{cyc}(G_X, G_Y) = \mathbb{E}_{x\sim p_x(x)}\{\|G_X(G_Y(x)) - x\|_1\} \\ + \mathbb{E}_{y\sim p_y(y)}\{\|G_Y(G_X(y)) - y\|_1\},\end{aligned} \tag{4}$$

where $\|\cdot\|_1$ represents the pixel-wise l_1 loss (sum of absolute difference of each pixel between the input and the back-projected output). The CycleGAN uses pixel-wise l_1 loss rather the l_2 loss since the former one encourages less blurring effect.

The final objective function of the CycleGAN can be written as the sum of (3) and (4):

$$\begin{aligned}\mathcal{L}_{cyclegan}(\tilde{G}, \tilde{D}) = \mathcal{L}_{X\to Y}(G_Y, D_Y) \\ + \mathcal{L}_{Y\to X}(G_X, D_X) \\ + \lambda\mathcal{L}_{cyc}(G_X, G_Y),\end{aligned} \tag{5}$$

where $\tilde{G} = (G_X, G_Y)$ and $\tilde{D} = (D_X, D_Y)$. $\lambda > 0$ controls the balance of the different objectives.

2.3. CGFCN

We build our model on top of the CycleGAN. Our model consists of five networks: G_X, G_Y, D_X, D_Y and F, where the (G_X, G_Y, D_X, D_Y) correspond to a CycleGAN, and the F is a standard FCN based image segmentation network. Fig. 2 shows an overview of the proposed method.

The goals of the proposed CGFCN is twofold. On one hand, we aim to learn two mappings $G_Y(x)$ and $G_X(y)$, where the former one maps the data from X to Y and the latter one maps the data from Y to X. On the other hand, we aim to train the F to predict pixel-wise building masks on the transformed data $G_Y(x)$. Since the CycleGAN can convert the source data to the target style while keeping their content unchanged, we use it to generate target-like images. In this way, the transformed data $G_Y(x)$ is given as the input of F and the ground truth of the original game data is given as the reference when training the segmentation network.

Suppose $\hat{x} \in \{0, 1\}$ represents the pixel-wise binary label of the image x, where "1" represents the pixel belonging to the category of "building" and "0" represent the pixel belonging to the category of "background". As the segmentation is essentially a pixel-wise binary classification process, we design the loss function of the segmentation network F as a standard pixel-wise binary cross-entropy loss. We express it as follows:

$$\begin{aligned}\mathcal{L}_{seg}(F, G_Y) = -\mathbb{E}_{(x,\hat{x})\sim p(x,\hat{x})}\{\hat{x}\log(F(G_Y(x))) \\ + (1 - \hat{x})\log(1 - F(G_Y(x)))\}.\end{aligned} \tag{6}$$

On combining the CycleGAN's objectives with the above segmentation loss, the final objective function of our method can be written as follows:

$$\begin{aligned}\mathcal{L}_{CGFCN}(\tilde{G}, \tilde{D}, F) = \mathcal{L}_{cyclegan}(\tilde{G}, \tilde{D}) \\ + \mu\mathcal{L}_{seg}(F, G_Y)\end{aligned} \tag{7}$$

where $\mu > 0$ controls the balance between the image translation task and the segmentation task. The training of our model can be considered as a minimax optimization process where the $\tilde{G}$ and F try to minimize its objective while the $\tilde{D}$ tries to maximize it:

$$\tilde{G}^\star, \tilde{D}^\star, F^\star = \min_{\tilde{G},F}\max_{\tilde{D}}\mathcal{L}_{CGFCN}(\tilde{G}, \tilde{D}, F). \tag{8}$$

Dataset	**Synthetic Remote Sensing Dataset**	**Real Remote Sensing Dataset**
Image Source	GTA-V Game Map (Los Santos)	Massachusetts Building [48]
Number of Images	121	150 (a subset)
Image Size	500 × 500 pixel	500 × 500 pixel
Resolution	∼1.0 m/pixel	1.0 m/pixel
Eval Split	Trainig/Evaluation Set	Testing Set

Table 1. The details of our experimental datasets. We train our model on the synthetic remote sensing dataset (GTA-V game map) and then run evaluation on the real remote sensing dataset (Massachusetts Building).

Since all networks of our model are differentiable, the image segmentation network F can be jointly trained with the CycleGAN networks in an end-to-end fashion.

A complete optimization pipeline of our method is summarized as follows:

- **Step 1.** Initialize the weights of the networks ($\tilde{G}$, $\tilde{D}$) with random initialization. Initialize the F using the ImageNet pre-train weights.
- **Step 2.** Fix $\tilde{G}$ and $\tilde{D}$, and update F to minimize the $\mathcal{L}_{CGFCN}$.
- **Step 3.** Fix F and $\tilde{D}$, and update $\tilde{G}$ to minimize the $\mathcal{L}_{CGFCN}$.
- **Step 4.** Fix F and $\tilde{G}$, and update $\tilde{D}$ to maximize the $\mathcal{L}_{CGFCN}$.
- **Step 5.** Repeat the steps 2-4 until the maximum epoch number reached.

2.4. Implementation Details

We build our generators $\tilde{G}$ and discriminators $\tilde{D}$ by following the configurations of the CycleGAN paper [44]. We build the $\tilde{D}$ as a local perception network - which only penalizes the image structures at the scale of patches (a.k.a the Markovian discriminator or "PatchGAN"). The $\tilde{D}$ tries to classify if each $N \times N$ patch in an image is a clean image (real) or a decomposed one (fake). This type of architecture can be equivalently implemented by building a fully convolutional network with $N \times N$ perceptive fields. Such design is more computationally efficient since the responses of all patches can be obtained by taking only one time of forward-propagation. We build the $\tilde{G}$ by following the configuration of the UNet [49]. We add skip connections to our separator between the layer i and layer $n - i$ for learning both high-level semantics and low-level details.

Our segmentation network F is built based on the ResNet-50 [6] by removing its fully connected layers and adding an additional 1×1 convolution layer at its output end. In this way, the network F can proceed an input image with an arbitrary size and aspect ratio. Besides, to increase the output resolution, we change the convolutional stride from 2 to 1 at the "Conv_3" layer and "Conv_4" layer in ResNet-50 to enlarge the output resolution from 1/32 to 1/8 of the input.

During the training, the $\tilde{D}$, $\tilde{G}$, and F are alternatively updated. The maximum training iteration is set to 200 epochs. We train $\tilde{D}$ $\tilde{G}$, and F by using Adam optimizer [61]. The $\tilde{D}$ and $\tilde{G}$ are trained from scratch. For the first 100 epochs, we set learning_rate $= 0.0001$. For the rest epochs, we reduce the learning rate to its 1/10. The F is trained from the ImageNet [62] pre-trained initialization with the learning rate of $1e^{-3}$. The learning rate decays to 90% per 10 epochs. We set $\lambda = 10.0$ and $\mu = 1.0$. To increase the diversity of the training images, the data augmentation is used during the training, including the random image rotation (90°, 180°, 270°) and vertical / horizontally image flipping.

3. Dataset and Evaluation Metrics

We build our aerial view image segmentation dataset based on the game map of the PC game GTA-V. We use a sub-region of the rendered satellite map as our training data. This part of the map is

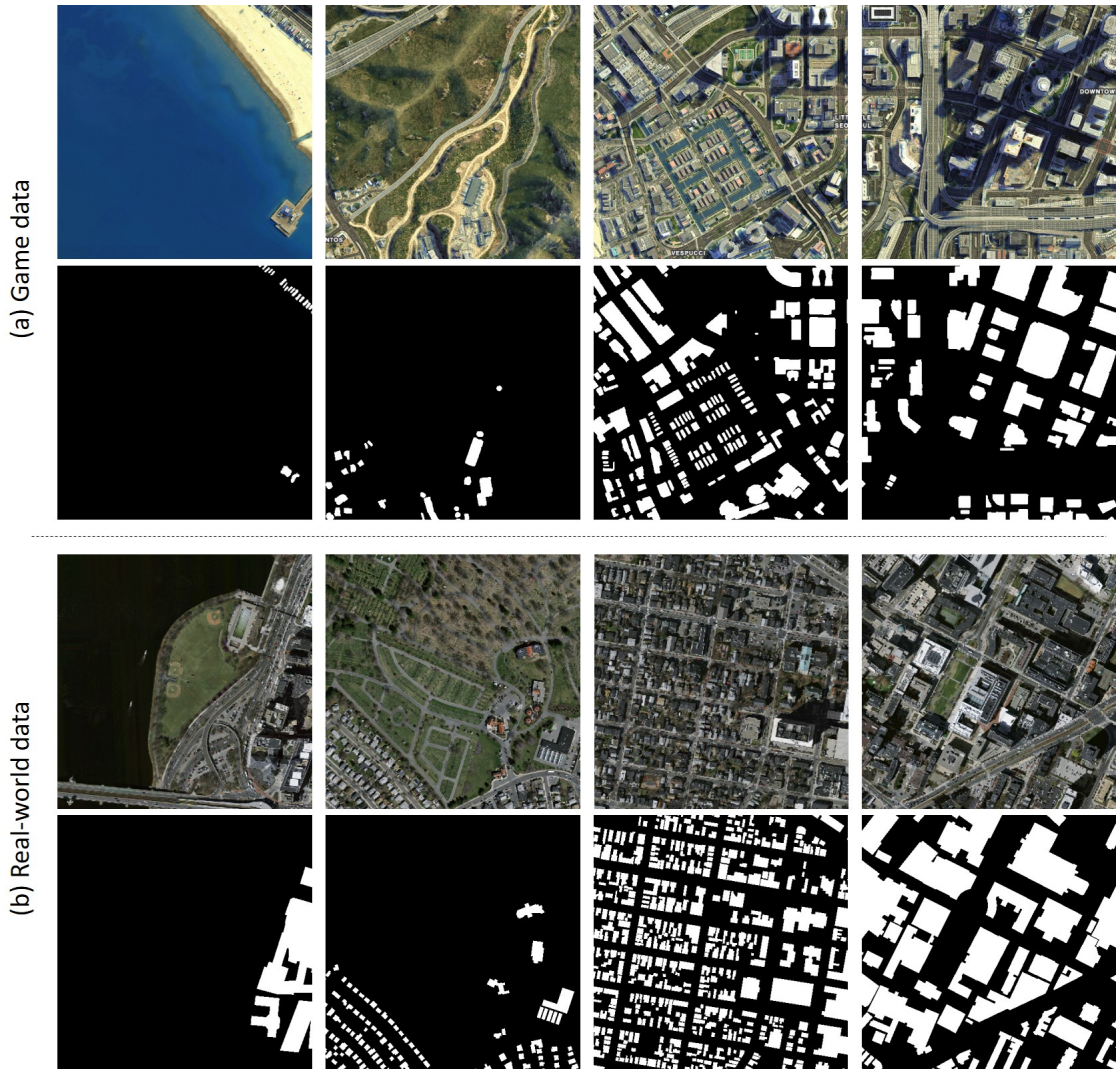

Figure 3. A preview of our two experimental datasets. The first two rows show some representative images and their ground truth labels from our synthetic dataset (our training and validation set). The last two rows show some images pairs from the real remote sensing dataset [48] (our testing set).

located in the urban part of the fictional city "Los Santos". We build its ground truth map based on its official legend (8,000×8,000 pixels) by manually annotating the building regions. As the GTA-V official map contains a Google map fashion color legend for various ground features, the manual annotation can be very efficient - it only takes half an hour for a single person to complete the annotation. Our dataset covers the most ground features of a typical coastal city, e.g., building, road, green-land, mountain, beach, harbor, wasteland, etc. In Fig. 3, the first two rows show some representative samples and their ground truth of our synthetic dataset.

We test our model on a real-world remote sensing dataset, the Massachusetts building detection dataset [48]. As the CycleGAN focuses on reducing image style differences between two sets of images, which requires the two datasets to have similar contents. Therefore, we use a subset of the Massachusetts building dataset as our test set where the images in this subset are captured above the urban area. All images in both our training set and test set are cropped to image slices with 500×500 pixel size and with the resolution of $\sim$1m/pixel before feeding into our networks. Table 1 gives the statistics of our experimental datasets.

We use the Intersection Over Union (IOU) as our evaluation metric for the segmentation results. The IOU metric is commonly used in previous building segmentation literature [63,64]. Given a

Figure 4. (Better viewed in color) Some image translation results by using our method. The first two rows show the translation results from the game domain to the real-world domain. The last two rows show the inverse translation results from the real-world domain to the game domain.

segmentation output and a ground truth reference image with the same size, the IOU is defined as follows:

$$\text{IOU} = \frac{N_{TP}}{N_{TP} + N_{FP} + N_{FN}}, \tag{9}$$

where N_{TP}, N_{FP} and N_{FN} represent the number of true positive, false positive and false negative pixels of the segmentation result.

4. Experimental Results

In this section, we first compare our method with some other state of the art segmentation methods. Then the ablation experiment is made to evaluate the effectiveness of each of our technical components. Finally, some additional controlled experiments are made to investigate whether the integration of semantic labels helps style conversion.

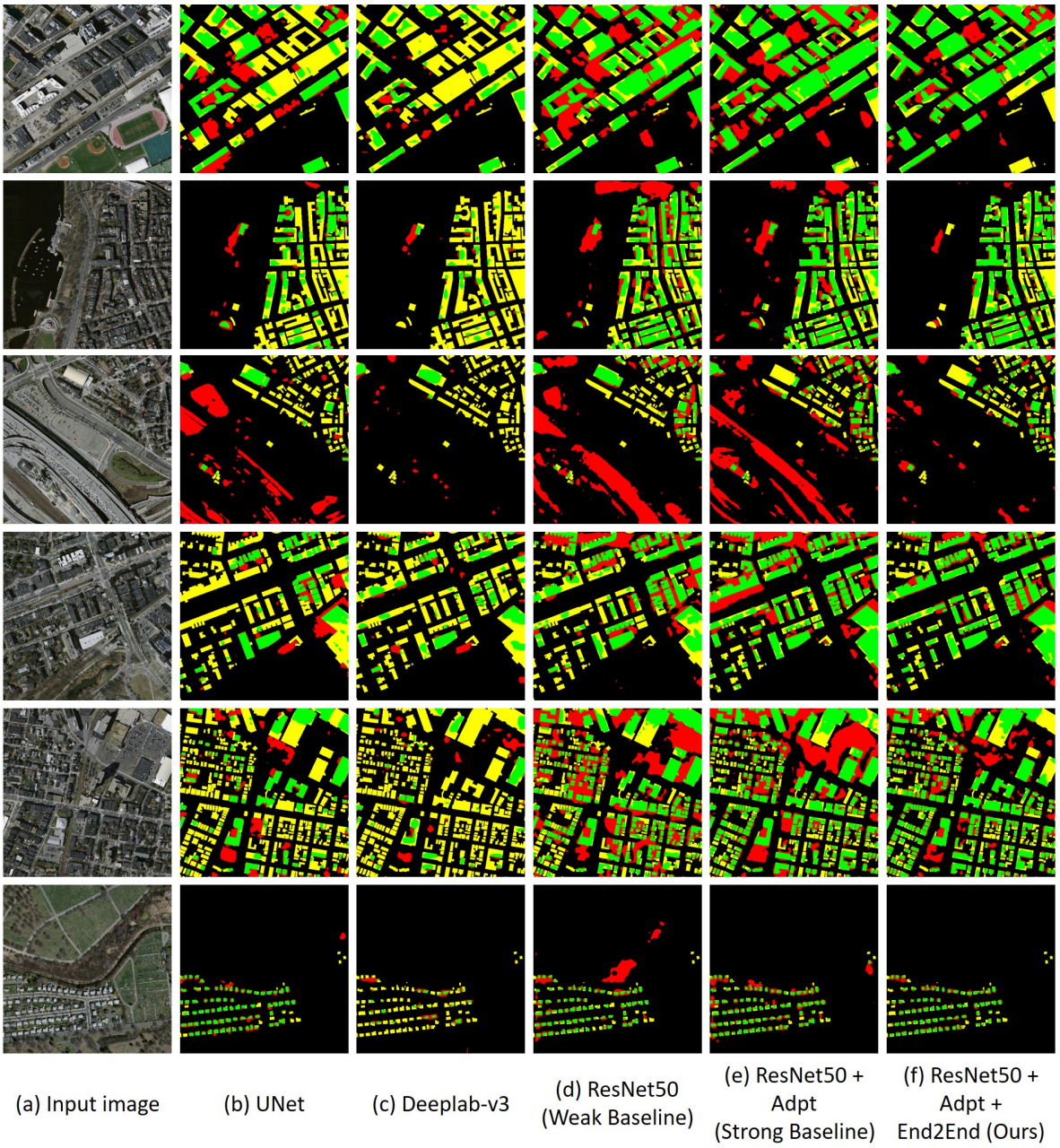

Figure 5. (Better viewed in color) Some building segmentation results of different methods: UNet [49], Deeplab-v3 [47], ResNet50-FCN (our weak baseline), ResNet50-FCN + Adapt (our strong baseline), and ResNet50-FCN + Adapt + End2End (our full implementation). Green pixels: true positives. Yellow pixels: false negatives. Red pixels: false positives.

Metric	UNet [49]	Deeplab-v3 [47]	CGFCN (Ours)
IOU (Test 1)	0.1592	0.1822	0.5218
IOU (Test 2)	0.2623	0.1715	0.5253
IOU (Test 3)	0.2837	0.1562	0.5220
IOU (Test 4)	0.2175	0.2022	0.5042
IOU (Test 5)	0.2586	0.2025	0.5355
Average	0.2363	0.1829	**0.5218**
Stdev (±)	0.0441	0.0179	**0.0101**

Table 2. A Comparison of different methods that are trained based on synthetic data and then tested on real data: UNet [49], Deeplab-v3 [47], and CGFCN (Ours). For each of these methods, we repeat the training of each model for five times and then record the accuracy of each model on our test set (marked as "Test-1" $\sim$ "Test-5"). The CGFCN obtains the best results in terms of both mean accuracy and stability.

4.1. Comparison with Other Methods

We compare our model with some state of the art semantic segmentation models, including Deeplab-v3 [47] and UNet [49]. These models are first trained on our training set and then directly evaluated on our test set without the help of the style transfer. All models are fully optimized for a fair comparison. For each of these methods, we repeat the training of each model for five times, and then record the accuracy of each model on our test set (marked as "Test-1" $\sim$ "Test-5").

Table 2 shows their accuracy during the five repeated tests. It should be noticed that although we do not apply any other tricks (e.g., feature fusion and dilated convolution) to increase the feature resolution, as those are used in UNet [49] and Deeplab-v3 [47], our method still achieves the best results in terms of both mean accuracy and stability (standard deviation).

Fig. 4 shows some image translation examples of our method, where the first two rows show some rendered game images and the "game $\rightarrow$ real world" translation results. The second two rows show some real world images from the Massachusetts building dataset and the "real world $\rightarrow$ game" translation results. It can be seen that the style of these images has been transformed to another domain while their contents are retained at the same time.

4.2. Ablation Analysis

To further evaluate the effectiveness of the proposed methods, the ablation experiment is conducted to analyze the importance of each component of the proposed method, including the "domain adaptation" (Adaptation) and the "end-to-end training" (End-to-End). For a fair comparison, we set our method and its variants with the same experimental configurations in data augmentation, and use the same training hyper-parameters. We first compare with a weak baseline method "ResNet50-FCN" where our segmentation network F is only trained according to Eq (6) without the help of adversarial domain adaptation (the first row in Table 3). Then, we gradually add other technical components.

- **Res50-FCN**: We train our segmentation network F according to Eq (6). The training is performed on game data and then the evaluation is performed on real data without the help of domain adaption (our weak baseline).
- **Adaptation**: We first train a CycleGAN model separately to transform the game data to the real-world data style. Then we train our segmentation network F based on the transformed data by freezing the parameters of the CycleGAN part (our strong baseline).

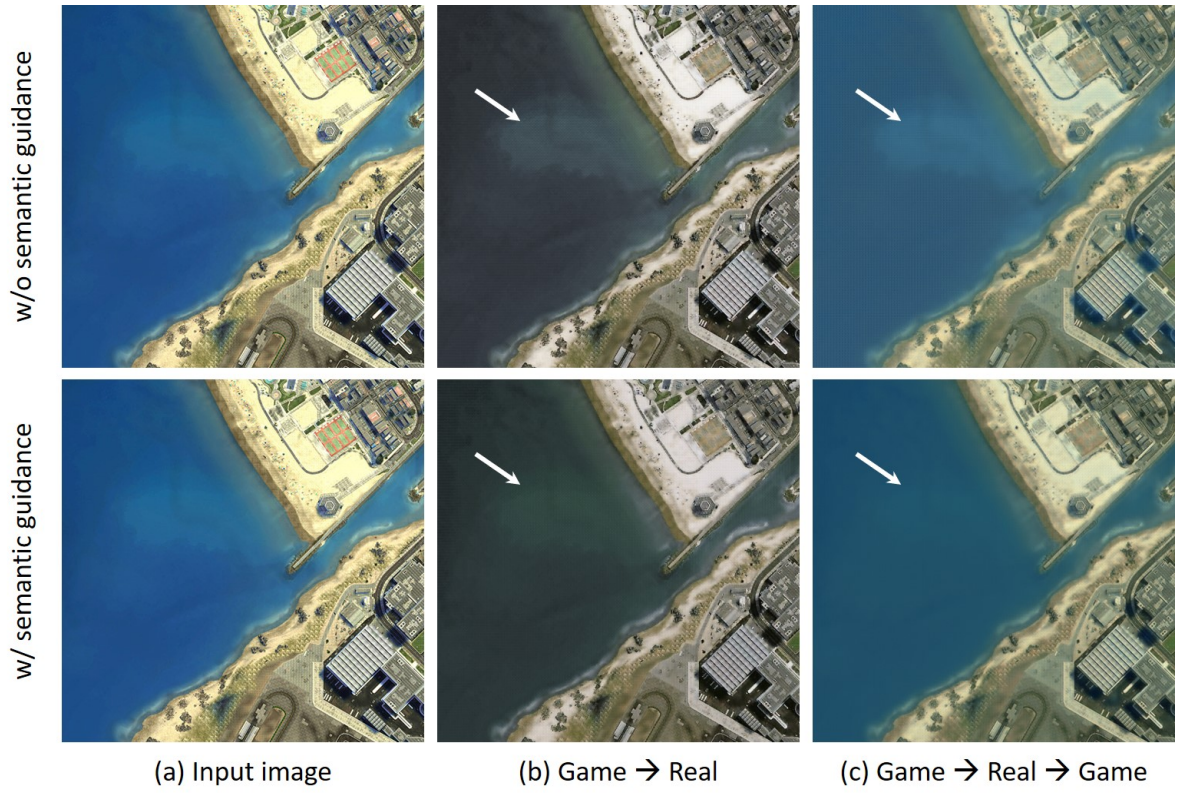

Figure 6. (Better viewed in color) A visual comparison of style transfer results with different configurations. The first row corresponds to the results of the original CycleGAN [44] and the second row corresponds to the results of our method. Columns: (a) input game data, (b) game → real, (c) game →real→game. It can be seen that the images generated by our method are closer to the target domain and thus helps cross-domain segmentation. It removes some unrelated image contents (e.g., the pollution area marked by the arrows) during the conversion.

- **End-to-End**: we jointly train the CycleGAN and our segmentation network F according to Eq (8) in an end-to-end fashion (our full implementation).

Table 3 shows the evaluation results of all the above variants. We can see the integration of domain adaptation and end-to-end learning yields noticeable improvements in the segmentation accuracy. Fig. 5 shows some building segmentation results of the UNet [49], Deeplab-v3 [47], and all the above-mentioned ablation variants. The green, yellow, and red pixels represent "true positives", "false negatives", and "false positives", respectively. Although the style transfer (Res50FCN+Adaptation, our strong baseline) improves the segmentation result, it still has some limitations. As shown in the third row of Fig.5, the flyover is falsely labeled as building by the UNet, ResNet50, and ResNet50-Adpt, while our end-to-end model (full-implementation) can effectively remove most false-alarms.

This improvement ($\sim$ 4%) is mainly owing to the introduction of semantic information, which benefits our method in generating more precisely stylized results. This indicates that the integration of the semantic information to the style transfer process helps to reduce the style difference between two datasets and thus a semantic segmentation model jointly trained with a style transfer model yields incremental segmentation results. Another reason for the improvement is due to the perturbation of the data introduced by the end-to-end training process, where the intermediate results produced by the CycleGAN produces small input variations to the segmentation network. This variation can be considered as a data augmentation process, which helps improve the generalization ability.

Ablations			Segmentation Accuracy (IOU) on Game Dataset					
Res50FCN	Adaptation	En2En	Test 1	Test 2	Test 3	Test 4	Test 5	Average ± Stdev
✓			0.4322	0.4146	0.4299	0.4342	0.4427	0.4307 ± **0.0091**
✓	✓		0.4708	0.4905	0.4753	0.4991	0.4785	0.4828 ± 0.0104
✓	✓	✓	0.5218	0.5253	0.5220	0.5042	0.5355	**0.5218** ± 0.0101

Table 3. Results of our ablation analysis on "domain adaptation" and "end-to-end training". Baseline method: Res50FCN. Adaptation: style transfer networks ($\tilde{D}$, $\tilde{G}$) and segmentation network F are separately trained. En2En: jointly train all networks in an end-to-end fashion. The integration of the domain adaptation and end-to-end learning yields noticeable improvements in the segmentation accuracy.

4.3. Do Semantic Labels Help Style Conversion?

Another advantage of our end-to-end training framework is that it introduces semantic information to the style transfer framework and thus it will benefit to style conversion. Fig. 6 gives a comparison example with or without the help of semantic guidance when performing the CycleGAN style conversion. There are subtle differences in the results produced by using the two configurations. The stylized images generated by our end-to-end trained CGFCN are much closer to the distribution of the target domain than that of the original CycleGAN [44]. This improvement helps in generating more accurate segmentation results.

To further evaluate the effectiveness of our method, we quantitatively compare with CycleGAN on their generated images, as shown in Table 4. We use three image similarity evaluation metrics: the Peak Signal-to-Noise Ratio (PSNR), the Structural Similarity (SSIM) index [65], and the Fréchet Inception Distance (FID) [66]. The PSNR and SSIM two are classic metrics for evaluating image restoration results. The FID is a more recent popular metric that can better evaluate the visual perceptual quality. The FID measures the deviation between the distribution of deep features of generated images and that of real images, which is widely used in adversarial image synthesis. It should be noticed that although the PSNR and SSIM are computed by comparing the resulting image to a reference image, which requires paired inputs, the FID can be evaluated free from such restrictions. As there is no ground truth reference in the "Game→Real" experiment, we only report the FID score in Table 4. To do this, we randomly divide the generated results and the real data into five groups and then compute the average FID similarity of them. We evaluate the style conversion results of two settings: 1) "game → real-world" conversion, and 2) "game → real-world → game" conversion. Our method achieves the best conversion results in terms of all evaluation metrics.

5. Conclusion

We investigate an interesting question that whether game data generalize well for remote sensing image segmentation. To do this, we training a deep learning model on the city map of the game "GTA-V" and then adapting the model to real-world remote sensing building segmentation tasks. To tackle the "domain shift" problem, we propose a CycleGAN-based FCN model where the mappings between the two domains are jointly learned with the building segmentation network. By using the above methods, we have obtained promising results. Experimental results suggest the effectiveness of our method for both segmentation and style conversion.

Author Contributions: Conceptualization, Z.S.; Methodology, Z.Z (Zhengxia Zou) and T.S.; Validation, T.S.; Formal Analysis, Z.S., Z.Z. (Zhengxia Zou), and Z.Z. (Zhou Zhang) ; Writing and original draft preparation: Z.Z. (Zhengxia Zou) and T.S.; Writing–Review and Editing, Z.S., T.S., Z.Z. (Zhou Zhang), and Z.Z. (Zhengxia Zou).

Funding: The work was supported by the National Key R&D Program of China under the Grant 2017YFC1405605, the National Natural Science Foundation of China under the Grant 61671037, the Beijing Natural Science

Metric	Game→Real FID	Game→Real→Game FID	PSNR	SSIM
CycleGAN	0.1893	0.1078	20.575	0.7214
CGFCN	**0.1621**	**0.0724**	**20.598**	**0.8038**
Game Data	0.1808	0.0000	$+\infty$	1.0000
Reference	*Real Data*	*Game Data*		

Table 4. Evaluation results on image style transfer results with different similarity evaluation metrics: FID [66], PSNR, and SSIM [65]. The column "Game→Real": we compute the similarity between a group of the real images and generated ones. The column "Game→Real→Game": we back-convert the generated image to the game domain and then compute their "self-similarity". For FID, lower scores indicate better. For PSNR (dB) and SSIM, higher scores indicate better. The CGFCN (ours) achieves the best results, which suggests that introducing semantical supervision will help improve image style conversion.

Foundation under the Grant 4192034 and the National Defense Science and Technology Innovation Special Zone Project.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *nature* **2015**, *521*, 436.
2. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep learning*; MIT press, 2016.
3. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems, 2012, pp. 1097–1105.
4. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* **2014**.
5. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the inception architecture for computer vision. Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 2818–2826.
6. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.
7. Zou, Z.; Shi, Z.; Guo, Y.; Ye, J. Object Detection in 20 Years: A Survey. *arXiv preprint arXiv:1905.05055* **2019**.
8. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. Proceedings of the IEEE conference on computer vision and pattern recognition, 2014, pp. 580–587.
9. Girshick, R. Fast r-cnn. Proceedings of the IEEE international conference on computer vision, 2015, pp. 1440–1448.
10. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems, 2015, pp. 91–99.
11. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 779–788.
12. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. European conference on computer vision. Springer, 2016, pp. 21–37.
13. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 3431–3440.
14. Noh, H.; Hong, S.; Han, B. Learning deconvolution network for semantic segmentation. Proceedings of the IEEE international conference on computer vision, 2015, pp. 1520–1528.
15. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs. ICLR, 2015.

16. Karpathy, A.; Fei-Fei, L. Deep visual-semantic alignments for generating image descriptions. Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 3128–3137.
17. Xu, K.; Ba, J.; Kiros, R.; Cho, K.; Courville, A.; Salakhudinov, R.; Zemel, R.; Bengio, Y. Show, attend and tell: Neural image caption generation with visual attention. International conference on machine learning, 2015, pp. 2048–2057.
18. Wu, Q.; Shen, C.; Wang, P.; Dick, A.; van den Hengel, A. Image captioning and visual question answering based on attributes and external knowledge. *IEEE transactions on pattern analysis and machine intelligence* **2017**, *40*, 1367–1381.
19. Dong, C.; Loy, C.C.; He, K.; Tang, X. Image super-resolution using deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence* **2015**, *38*, 295–307.
20. Kim, J.; Kwon Lee, J.; Mu Lee, K. Accurate image super-resolution using very deep convolutional networks. Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 1646–1654.
21. Ledig, C.; Theis, L.; Huszár, F.; Caballero, J.; Cunningham, A.; Acosta, A.; Aitken, A.; Tejani, A.; Totz, J.; Wang, Z.; others. Photo-realistic single image super-resolution using a generative adversarial network. Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 4681–4690.
22. Zhang, L.; Zhang, L.; Du, B. Deep learning for remote sensing data: A technical tutorial on the state of the art. *IEEE Geoscience and Remote Sensing Magazine* **2016**, *4*, 22–40.
23. Zou, Z.; Shi, Z. Random access memories: A new paradigm for target detection in high resolution aerial remote sensing images. *IEEE Transactions on Image Processing* **2018**, *27*, 1100–1111.
24. Shi, Z.; Zou, Z. Can a machine generate humanlike language descriptions for a remote sensing image? *IEEE Transactions on Geoscience and Remote Sensing* **2017**, *55*, 3623–3634.
25. Penatti, O.A.; Nogueira, K.; Dos Santos, J.A. Do deep features generalize from everyday objects to remote sensing and aerial scenes domains? Proceedings of the IEEE conference on computer vision and pattern recognition workshops, 2015, pp. 44–51.
26. Xia, G.S.; Bai, X.; Ding, J.; Zhu, Z.; Belongie, S.; Luo, J.; Datcu, M.; Pelillo, M.; Zhang, L. DOTA: A large-scale dataset for object detection in aerial images. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 3974–3983.
27. Ros, G.; Sellart, L.; Materzynska, J.; Vazquez, D.; Lopez, A.M. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 3234–3243.
28. Richter, S.R.; Vineet, V.; Roth, S.; Koltun, V. Playing for data: Ground truth from computer games. European conference on computer vision. Springer, 2016, pp. 102–118.
29. Patel, V.M.; Gopalan, R.; Li, R.; Chellappa, R. Visual domain adaptation: A survey of recent advances. *IEEE signal processing magazine* **2015**, *32*, 53–69.
30. Fernando, B.; Habrard, A.; Sebban, M.; Tuytelaars, T. Unsupervised visual domain adaptation using subspace alignment. Proceedings of the IEEE international conference on computer vision, 2013, pp. 2960–2967.
31. Wang, M.; Deng, W. Deep visual domain adaptation: A survey. *Neurocomputing* **2018**, *312*, 135–153.
32. Yao, T.; Pan, Y.; Ngo, C.W.; Li, H.; Mei, T. Semi-supervised domain adaptation with subspace learning for visual recognition. Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 2142–2150.
33. Zhang, Y.; Qiu, Z.; Yao, T.; Liu, D.; Mei, T. Fully Convolutional Adaptation Networks for Semantic Segmentation. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 6810–6818.
34. Hoffman, J.; Tzeng, E.; Park, T.; Zhu, J.Y.; Isola, P.; Saenko, K.; Efros, A.A.; Darrell, T. Cycada: Cycle-consistent adversarial domain adaptation. *arXiv preprint arXiv:1711.03213* **2017**.
35. Tzeng, E.; Hoffman, J.; Saenko, K.; Darrell, T. Adversarial discriminative domain adaptation. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 7167–7176.
36. Sankaranarayanan, S.; Balaji, Y.; Jain, A.; Nam Lim, S.; Chellappa, R. Learning from synthetic data: Addressing domain shift for semantic segmentation. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 3752–3761.
37. Gatys, L.A.; Ecker, A.S.; Bethge, M. Image Style Transfer Using Convolutional Neural Networks. Computer Vision and Pattern Recognition, 2016, pp. 2414–2423.

38. Lu, M.; Zhao, H.; Yao, A.; Xu, F.; Chen, Y.; Zhang, L. Decoder network over lightweight reconstructed feature for fast semantic style transfer. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2469–2477.
39. Li, Y.; Fang, C.; Yang, J.; Wang, Z.; Lu, X.; Yang, M.H. Universal style transfer via feature transforms. Advances in Neural Information Processing Systems, 2017, pp. 386–396.
40. Liao, J.; Yao, Y.; Yuan, L.; Hua, G.; Kang, S.B. Visual attribute transfer through deep image analogy. *ACM Transactions on Graphics (TOG)* **2017**, *36*, 120.
41. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. Advances in neural information processing systems, 2014, pp. 2672–2680.
42. Goodfellow, I. NIPS 2016 tutorial: Generative adversarial networks. *arXiv preprint arXiv:1701.00160* **2016**.
43. Lu, Y.; Tai, Y.W.; Tang, C.K. Attribute-Guided Face Generation Using Conditional CycleGAN. The European Conference on Computer Vision (ECCV), 2018.
44. Zhu, J.Y.; Park, T.; Isola, P.; Efros, A.A. Unpaired Image-To-Image Translation Using Cycle-Consistent Adversarial Networks. The IEEE International Conference on Computer Vision (ICCV), 2017.
45. Chang, H.; Lu, J.; Yu, F.; Finkelstein, A. PairedCycleGAN: Asymmetric Style Transfer for Applying and Removing Makeup. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.
46. Inoue, N.; Furuta, R.; Yamasaki, T.; Aizawa, K. Cross-domain weakly-supervised object detection through progressive domain adaptation. Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 5001–5009.
47. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence* **2018**, *40*, 834–848.
48. Mnih, V. *Machine learning for aerial image labeling*; University of Toronto (Canada), 2013.
49. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. International Conference on Medical image computing and computer-assisted intervention. Springer, 2015, pp. 234–241.
50. Denton, E.L.; Chintala, S.; Fergus, R.; others. Deep generative image models using a laplacian pyramid of adversarial networks. Advances in neural information processing systems, 2015, pp. 1486–1494.
51. Radford, A.; Metz, L.; Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434* **2015**.
52. Zhang, H.; Goodfellow, I.; Metaxas, D.; Odena, A. Self-attention generative adversarial networks. *arXiv preprint arXiv:1805.08318* **2018**.
53. Brock, A.; Donahue, J.; Simonyan, K. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096* **2018**.
54. Isola, P.; Zhu, J.Y.; Zhou, T.; Efros, A.A. Image-to-image translation with conditional adversarial networks. Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 1125–1134.
55. Choi, Y.; Choi, M.; Kim, M.; Ha, J.W.; Kim, S.; Choo, J. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 8789–8797.
56. Li, J.; Liang, X.; Wei, Y.; Xu, T.; Feng, J.; Yan, S. Perceptual generative adversarial networks for small object detection. IEEE CVPR, 2017.
57. Bai, Y.; Zhang, Y.; Ding, M.; Ghanem, B. SOD-MTGAN: Small Object Detection via Multi-Task Generative Adversarial Network. *Computer Vision-ECCV* **2018**, pp. 8–14.
58. Wang, X.; Shrivastava, A.; Gupta, A. A-fast-rcnn: Hard positive generation via adversary for object detection. IEEE Conference on Computer Vision and Pattern Recognition, 2017.
59. Wang, X.; Yu, K.; Wu, S.; Gu, J.; Liu, Y.; Dong, C.; Qiao, Y.; Change Loy, C. Esrgan: Enhanced super-resolution generative adversarial networks. Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 0–0.
60. Yuan, Y.; Liu, S.; Zhang, J.; Zhang, Y.; Dong, C.; Lin, L. Unsupervised image super-resolution using cycle-in-cycle generative adversarial networks. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2018, pp. 701–710.
61. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* **2014**.

62. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Fei-Fei, L. Imagenet: A large-scale hierarchical image database. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Ieee, 2009, pp. 248–255.
63. Ji, S.; Wei, S.; Lu, M. Fully Convolutional Networks for Multisource Building Extraction from an Open Aerial and Satellite Imagery Data Set. *IEEE Transactions on Geoscience and Remote Sensing* **2019**, *57*, 574 – 586.
64. Maggiori, E.; Tarabalka, Y.; Charpiat, G.; Alliez, P. Convolutional Neural Networks for Large-Scale Remote-Sensing Image Classification. *IEEE Transactions on Geoscience and Remote Sensing* **2017**, *55*, 645 – 657.
65. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P.; others. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing* **2004**, *13*, 600–612.
66. Heusel, M.; Ramsauer, H.; Unterthiner, T.; Nessler, B.; Hochreiter, S. Gans trained by a two time-scale update rule converge to a local nash equilibrium. Advances in Neural Information Processing Systems, 2017, pp. 6626–6637.