remote sensing

MDPI

Article

AE-GAN-Net: Learning Invariant Feature Descriptor to Match Ground Camera Images and a Large-Scale 3D Image-Based Point Cloud for Outdoor Augmented Reality

Weiquan Liu [1], Cheng Wang [1,*], Xuesheng Bian [1], Shuting Chen [2], Wei Li [1], Xiuhong Lin [1], Yongchuan Li [1], Dongdong Weng [3], Shang-Hong Lai [4] and Jonathan Li [1,5]

1 Fujian Key Laboratory of Sensing and Computing for Smart Cities, School of Informatics, Xiamen University, Xiamen 361005, China; wqliu@stu.xmu.edu.cn (W.L.); xsbian@stu.xmu.edu.cn (X.B.); 23020150150705@xmu.stu.edu.cn (W.L.); xhlinxm@stu.xmu.edu.cn (X.L.); liyongchuan@xmu.edu.cn (Y.L.); junli@xmu.edu.cn (J.L.)

2 Information Engineering School, Chengyi University College, Jimei University, Xiamen 361021, China; chenst2016@jmu.edu.cn

3 Beijing Engineering Research Center of Mixed Reality and Advanced Display, School of Optics and Photonics, Beijing Institute of Technology, Beijing 100081, China; crgj@bit.edu.cn

4 Department of Computer Science, National Tsing Hua University, Hsinchu 30013, Taiwan; lai@cs.nthu.edu.tw

5 Department of Geography and Environmental Management University of Waterloo, Waterloo, ON N2L 3G1, Canada

* Correspondence: cwang@xmu.edu.cn; Tel.: +86-592-2580003

Received: 3 August 2019 ; Accepted: 19 September 2019 ; Published: 26 September 2019

Abstract: Establishing the spatial relationship between 2D images captured by real cameras and 3D models of the environment (2D and 3D space) is one way to achieve the virtual–real registration for Augmented Reality (AR) in outdoor environments. In this paper, we propose to match the 2D images captured by real cameras and the rendered images from the 3D image-based point cloud to indirectly establish the spatial relationship between 2D and 3D space. We call these two kinds of images as cross-domain images, because their imaging mechanisms and nature are quite different. However, unlike real camera images, the rendered images from the 3D image-based point cloud are inevitably contaminated with image distortion, blurred resolution, and obstructions, which makes image matching with the handcrafted descriptors or existing feature learning neural networks very challenging. Thus, we first propose a novel end-to-end network, AE-GAN-Net, consisting of two AutoEncoders (AEs) with Generative Adversarial Network (GAN) embedding, to learn invariant feature descriptors for cross-domain image matching. Second, a domain-consistent loss function, which balances image content and consistency of feature descriptors for cross-domain image pairs, is introduced to optimize AE-GAN-Net. AE-GAN-Net effectively captures domain-specific information, which is embedded into the learned feature descriptors, thus making the learned feature descriptors robust against image distortion, variations in viewpoints, spatial resolutions, rotation, and scaling. Experimental results show that AE-GAN-Net achieves state-of-the-art performance for image patch retrieval with the cross-domain image patch dataset, which is built from real camera images and the rendered images from 3D image-based point cloud. Finally, by evaluating virtual–real registration for AR on a campus by using the cross-domain image matching results, we demonstrate the feasibility of applying the proposed virtual–real registration to AR in outdoor environments.

Keywords: 3D image-based point cloud; cross-domain image; image patch matching; invariant feature descriptor; outdoor augmented reality; deep learning

1. Introduction

AR is considered as a technology that overlays virtual objects (augmented components) onto the real world [1]. To date, AR, which recently has advanced greatly, demonstrates impressive performance in Artificial Intelligence (AI) applications, including areas in remote sensing [2–5], education [1,6,7], medicine [8–10], etc. Moreover, virtual–real registration is a critical technique and a fundamental problem for AR. The virtual–real registration process is defined as the superimposing of virtual objects onto a real scene using information extracted from the scene [11,12]. In detail, AR virtual–real registration is the degree to which 3D information is accurately placed and integrated as part of the real environment [13]. The objects in the real and 3D scene should be correctly aligned with respect to each other, or the illusion that the two co-exist is compromised [14]. Thus, the performance of AR strongly depends on the accuracy of virtual–real registration.

Nowadays, most AR virtual–real registration algorithms are normally applied to indoor environments; few are applied to outdoor environments. Uncontrolled factors and complexity of large scale data (such as hundreds of objects, dramatic changes in illumination, etc.) result in outdoor environments being uncontrolled scenes. Thus, it is very challenging to achieve virtual–real registration of AR in large-scale outdoor environments.

In this paper, based on the following two factors, we developed a novel solution for the virtual–real registration of AR in outdoor environments: (1) Aerial images, captured by Unmanned Aerial Vehicles (UAVs), provide sufficient data (vertical and oblique aerial images) for large-scale 3D image-based point clouds using Structure-from-Motion (SfM) [15,16] algorithms. Because of the development of low-cost UAVs and light weight imaging sensors in the past decade [17], this 3D image-based point cloud becomes feasible and provides basic 3D information for outdoor environments for AR applications. (2) Real images captured from ground mobile devices, called ground camera images, provide an important clue for estimating the initial location and orientation of a user in the 3D environment. The overview of the proposed AR virtual–real registration approach in outdoor environments is shown in Figure 1.

The four stages in the pipeline for the proposed virtual–real registration of AR in outdoor environments (Figure 1) are as follows: (1) Aerial images, captured by UAVs, are used to generate a 3D image-based point cloud in urban environments via SfM technology. (2) Camera pose, acquired from a Global Navigation Satellite System (GNSS) and Inertial Measurement Units (IMU) from mobile devices as an initial estimate, is used to synthesize an image, which is rendered from the same viewpoint as the 3D image-based point cloud (In this paper, we call this synthetic image as 'rendered image'). A schematic of the rendering process is shown within the green dashed bounding box in Figure 1. (3) A ground camera image and the rendered image are matched by using the invariant feature descriptors, which are learned by our proposed AE-GAN-Net. The details are discussed in the following Section. Then, the above matching results are used to indirectly infer the spatial relationship between the 3D image-based point cloud and ground camera image. The schematic of the image patch matching process is shown within the blue dashed bounding box in Figure 1. (4) Using the following strategy, virtual object (Spiderman in Figure 1) is registered to a real scene by: first, determine where the virtual object (Spiderman) is to be placed in the 3D image-based point cloud; second, by using the inferred spatial relationship between 3D and 2D space, the virtual object (Spiderman) is registered to the cellphone image.

In detail, the inference of the spatial relationship is shown within the red dashed bounding box in Figure 1. The 3D image-based point cloud, ground camera image, and rendered image are denoted as M, C_I, R_I, respectively; the projection matrix from the 3D image-based point cloud to the rendered image is denoted as P (P is positioning information obtained from the mobile devices). Intuitively, the relationship between the 3D image-based point cloud and the rendered image is $P \cdot M \rightarrow R_I$. Then, supposing the transformation relationship from the rendered image to the ground camera image is T, the result is $T \cdot R_I \rightarrow C_I$. Thus, the spatial relationship between the 3D image-based point cloud and the ground camera image (3D space and 2D space) is indirectly inferred as $T \cdot (P \cdot M) \rightarrow C_I$.

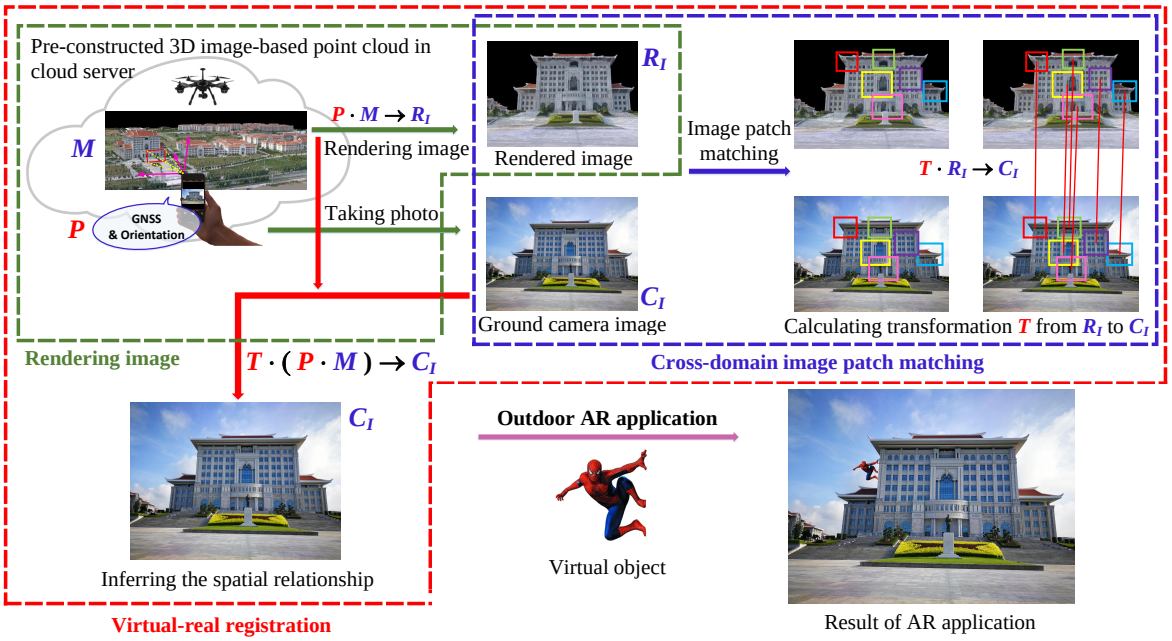
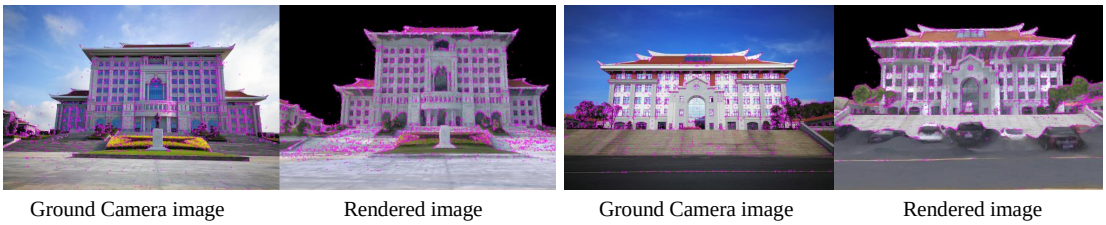


Figure 1. Overview of the proposed AR virtual–real registration approach in outdoor environments. Within the green dashed bounding box is the schematic of the synthetic image rendered from the 3D image-based point cloud; within the blue dashed bounding box is the cross-domain image patch matching by the invariant feature descriptors learned through the proposed AE-GNA-Net; within the red dashed bounding box is the virtual–real registration. The symbols represent the following: M: 3D image-based point cloud; C_I: Ground camera image; R_I: Synthetic image rendered from 3D image-based point cloud; P: Position information from mobile phone; T: Transformation relationship from R_I to C_I.

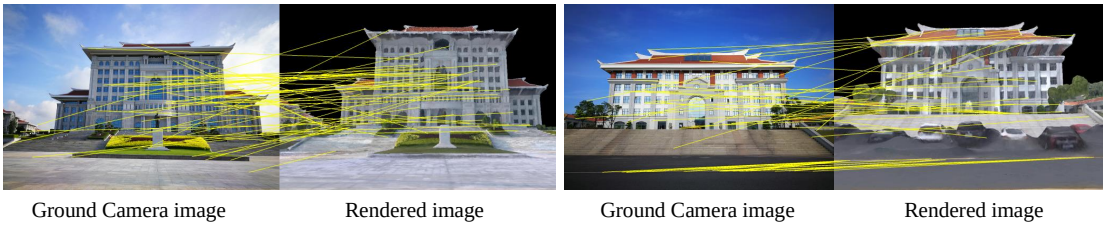
Essentially, the core problem for the proposed virtual real registration is to estimate the above assuming transformation matrix, T, from the rendered image to the ground camera image. Thus, our motivation is to indirectly establish the spatial relationship between 2D and 3D space by matching the ground camera images to the corresponding rendered images. It should be noted that the imaging mechanisms and nature of ground camera images and rendered images are different; thus, we consider them as cross-domain images.

However, for the following three reasons, it is challenging to match the ground camera images and the rendered images: (1) These two kinds of cross-domain images tend to have a gap in different local appearances. This characteristic data gap of different domains is the major difficulty for measuring the similarity of the features extracted from different domain images. (2) The rendered images from the 3D image-based point cloud have bias and drift from the corresponding ground camera images, because the position information from the mobile phone is an initial and coarse estimation. (3) It is difficult for the outdoor aerial images captured by UAVs to cover all terrain details (e.g., the structures under the eaves, the bottoms of buildings, and objects close to the ground). So the quality of the 3D image-based point cloud reconstructed by the SfM algorithm is inadequate. Therefore, rendered images, which have large distortion, low resolution, structural repetitiveness, and occlusions, are generally of low quality (See Figure 2). Thus, the image matching between ground camera images and rendered images is beyond the reach of the handcrafted features, such as Scale Invariant Feature Transform (SIFT) [18], Speeded Up Robust Features (SURF) [19], Efficient Dense Descriptor Applied to Wide-baseline Stereo (DAISY) [20], Oriented FAST and Rotated BRIEF (ORB) [21], etc. Some failed matching results are shown in Figure 3.

Figure 2. Detailed enlargements of a ground camera image and the corresponding rendered image from a 3D image-based point cloud. Top: Detailed enlargements of a ground camera image; Bottom: Detailed enlargements of a corresponding rendered image. Within the same colored bounding box are the two rows representing the corresponding paired image patches of the ground camera image and the corresponding rendered image.



(**a**) The extracted keypoints which used to calculate the SIFT descriptor



(**b**) Image matching results by SIFT

Figure 3. Examples of failed matching results between the ground camera images and the rendered images by SIFT. Left: Ground camera images; Right: Corresponding rendered images with the same viewpoint.

As seen in Figure 3a, the extracted keypoints of the ground camera images are robust. In contrast, a 3D image-based point cloud generated by the SfM algorithm has distortion and low resolution, which causes the surface of the 3D model to be coarse. Thus, the synthetic images rendered from the 3D image-based point cloud are low quality, and the texture of the rendered images is coarse. Consequently, the extracted keypoints of the rendered image are disorderly and unsystematic (see Figure 3a). Specifically, in Figure 3b, the final SIFT matching result proves that the SIFT descriptors, which are calculated by the keypoint of ground camera images and rendered images, are dissimilar.

Nowadays, deep neural networks are applied successfully to feature descriptors learning, such as Siamese [22–25] and triplet networks [26–30]. In addition, we are inspired by the idea that a SIFT descriptor is calculated in a local circular area (actually a blob) [31], which is essentially a patch-based framework. Thus, instead of using handcrafted keypoints, we consider matching the ground camera images and the corresponding rendered images by using the image patch matching strategy with a learning scheme.

Specifically, we consider cross-domain image matching to be a retrieval problem. First, a large number of rendered image patches are established as a retrieval database, and that number is

represented as a feature descriptor database. Second, the learned feature descriptors of the ground camera image patches are used to retrieve the matching rendered image patches from the database. Thus, our goal is to learn the invariant image patch feature descriptors for image matching between ground camera images and rendered images.

In this paper, we propose an end-to-end network, AE-GAN-Net, to learn robust invariant local patch feature descriptors for ground camera images and rendered images. AE-GAN-Net consists of two AutoEncoders (AEs) [32] with a shared decoder and an embedded Generative Adversarial Network (GAN) [33]. First, the two AEs extract feature descriptors of cross-domain image patch pairs; second, the GAN is embedded to improve information preservation in the feature encoder section. In training, labeled raw cross-domain image pairs are fed into AE-GAN-Net. Then, AE-GAN-Net is optimized by the introduced domain-consistent loss, which consists of content, feature consistency, and adversarial losses. The outputs of AE-GAN-Net are 128-dimensional compact descriptors. The Euclidean distances between two descriptors reflect patch similarity. In addition, several AR experimental applications are used to evaluate the possibility for the proposed AR virtual–real registration approach in an open, outdoor campus environment. The major contributions of this paper are as follows:

(1) We propose a novel network structure, AE-GAN-Net, to learn invariant feature descriptors for ground camera images and synthetic images rendered from 3D image-based point clouds. The learned feature descriptor is invariant against the changes on distortion, viewpoints, spatial resolution, rotation, and scale.
(2) The introduced domain-consistent loss simultaneously well preserves the image content and balances the feature consistency across the ground camera images and rendered images.
(3) The invariant feature descriptors learned by the proposed AE-GAN-Net achieve state-of-the-art image retrieval performance for ground camera images retrieved from the dataset formed by rendered images.

In summary, our research, which is problem-driven, focuses on exploring an effective solution for virtual–real registration of AR in outdoor environments. We propose AE-GAN-Net to learn invariant feature descriptors for cross-domain image matching between ground camera images and rendered images.

2. Related Work

2.1. Outdoor Augmented Reality

Recently, many effective AR virtual–real registration methods, which are based on visual fiducial markers [34–36], require controllable environments (usually indoor environments) and pre-placed markers. However, in outdoor environments, it is difficult to cover such an uncontrolled environment with markers. Besides, many outdoor AR approaches, such as outdoor military affairs, AR map navigation, etc., are usually combined with multiple sensors (such as GNSS receiver, gyroscope, inertial, and magnetic sensors, etc.) and vision-based methods for virtual–real registration [2,37]. However, these methods rely heavily on the accuracy of multiple sensor fusions. Such sensors suffer from many problems, e.g., deterioration in GNSS precision, deviation in gyroscope sensors, drift in the output of inertial and magnetic sensors. The above problems are challenging for virtual–real registration of AR in large-scale outdoor environments.

2.2. Feature Descriptors

For image patch matching problems, feature descriptors are designed to provide discriminative representations of salient image patches [38]. Importantly, the ideally designed feature descriptors should be robust against variations in viewpoint, illumination, photometric and geometric changes.

Handcrafted feature descriptors have reached maturity by the introduction of SIFT [18], SURF [19], DAISY [20], etc. SIFT computes local histograms of gradients to generate feature descriptors; whereas, SURF uses integral image representations to accelerate the computation. Following similar routes,

for extracting dense feature descriptors by convolving the maps of oriented gradients to approximate the histograms, DAISY yields large computational gains. Although handcrafted feature descriptors have performed extremely well, they are now outperformed by newer descriptors that are learned by deep neural networks.

Siamese [22–24,39,40] and triplet networks [27–30] are the mainstream network architectures to learn feature descriptors from raw image patches by training with large volumes of data. The Siamese network is a very popular and well-known deep neural network that uses the same weights, while working in tandem on two different input vectors, to compute comparable output vectors [41,42]. There are two branches in the Siamese network that share exactly the same architecture and the same set of weights to learn features. In detail, Siamese networks are divided into two categories by a metric network. MatchNet [43] and DeepCompare [44] are typical Siamese networks, which learn nonlinear distance metric for matching with a metric network. MatchNet and DeepCompare are highly accurate in binary decision, but are expensive cost in memory and computation. The obvious drawback of these networks is that the learned feature descriptors cannot apply Nearest Neighbor Search (NNS) for retrieval. On the contrary, Siamese networks, without metric network, learn invariant feature descriptors, which are used to replace directly the previous handcrafted descriptors. DeepDesc [22] outputs a 128-dimensional descriptor for an image by using margin-based contrastive loss to train the Siamese network. [39] and [40] proposed similar network frameworks with DeepDesc. Furthermore, there are several variants of the Siamese network, such as L2-Net [23], DeepCD [24], etc. Recently, triplet networks, such as TNet [27], DOAP [28], DDSAT [29], DescNet [30] etc., improve image patch matching performance. They use triplet loss [26] to learn more invariant feature descriptors than Siamese networks. Although achieving excellent performance in image matching on several datasets, such as Brown [45], Oxford [46], and Hpatches dataset [47], the above mentioned networks cannot achieve satisfying feature representations for the ground camera images and the rendered images from 3D image-based point clouds.

In addition, H-Net and H-Net++ [25] explore the matching problem for ground camera images and rendered images. On one hand, H-Net, by incorporating the AutoEncoder into the Siamese network, has excellent performance on the binary judgement of image patch matching on ground camera images and rendered images; however, the feature descriptors learned by H-Net cannot be used for retrieval. On the other hand, the feature descriptors learned by the improved H-Net++ can be used for retrieval, but the retrieval accuracy is insufficient, which leads to more mismatches during matching.

3. AE-GAN-Net

Recently, GANs have achieved state-of-the-art performance in the field of image generation producing very realistic images in an unsupervised setting [33,48,49]. There are two parts to GAN. One is the generative network (generator); the other is the discriminative network (discriminator). Typically, the adversarial strategy of GAN is that the generative network learns a mapper from a latent space to a data distribution of interest, while the discriminative network distinguishes candidates produced by the generator from the true data distribution. Thus, inspired by the above GANs approach, we propose AE-GAN-Net to learn the invariant feature descriptors for ground camera images and rendered images. AE-GAN-Net consists of two AutoEncoders with a shared decoder and one embedded GAN. The detailed framework of AE-GAN-Net is shown in Figure 4.

AE-GAN-Net aims to use an adversarial strategy to encourage the paired images generated by the generator (if the inputs are matching paired cross-domain images) to be as similar as possible, and let the paired images generated by the generator (if the inputs are non-matching paired cross-domain images) be dissimilar. Then, we use the shared decoder to backwardly infer that the learned feature descriptors are invariant for the inputs, which are matching pairs of cross-domain images.

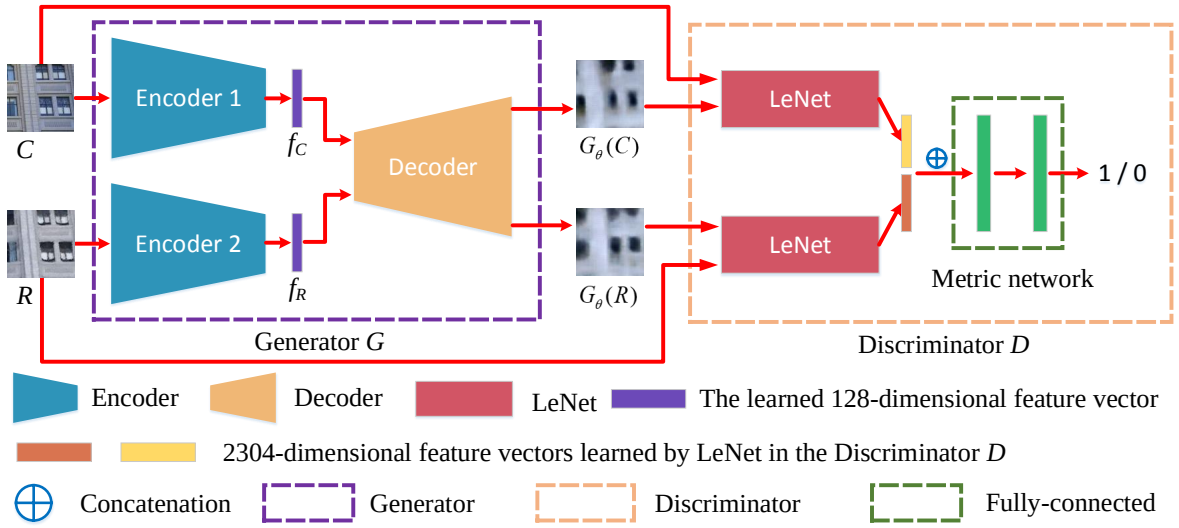


Figure 4. Our proposed AE-GAN-Net architecture.

3.1. Network Structure

Furthermore, as shown in Figure 4, AE-GAN-Net can be divided into two components: generator network *G* and discriminator network *D*. The descriptions of generator and discriminator are as follows:

Generator network *G*. Generator *G* consists of two autoencoders with a shared decoder. The inputs are labeled raw paired cross-domain image patches whose sizes are resized to 256*256*3, and the outputs are 128-dimensional feature descriptor vectors learned by the encoders. In detail, two unshared encoders have the same structure: convolution layers with zero padding and max pooling layers without zero padding. Batch Normalization (BN) [50] is used after each convolution; the non-linear activate function is SeLU [51]. The structure of the encoder is as follows: C(32,5,2)-BN-SeLU-C(64,5,2)-BN-SeLU-P(3,2)-C(96,3,1)-BN-SeLU-C(256,3,1)-BN-SeLU-P(3,2)-C(384,3,1)-BN-SeLU-C(384,3,1)-BN-SeLU-C(256,3,1)-BN-SeLU-P(3,2)-C(128,7,1)-BN-SeLU, where $\mathrm{C}(n;k;s)$ is a convolution layer with n filters of kernel size $k \times k$ having stride s; $\mathrm{P}(k;s)$ is the max pooling layer of size $k \times k$ with stride s.

For the shared decoder in generator *G*, we use transposed convolution to reconstruct the learned 128-dimensional feature descriptors as a 256*256*3 image. Before deconvolution (transposed convolution), we first map the 128-dimensional feature descriptor to a 1,024-dimensional vector by a fully connected layer. The detailed decoder structure is as follows: FC(128, 1024)-TC(128,4,2)-SeLU-TC(64,4,2)-SeLU-TC(32,4,2)-SeLU-TC(16,4,2)-SeLU-TC(8,4,2)-SeLU-TC(4,4,2)-SeLU-TC(3,4,2)-Sigmoid, where $\mathrm{FC}(p;q)$ represents the input p-dimensional feature vector map to q-dimensional feature vector through a fully connected layer; $\mathrm{TC}(n;k;s)$ represents the transposed convolution with n output channels of size $k \times k$ and stride s.

Discriminator network *D*. The input for discriminator *D* includes two kinds of cross-domain image pair patches with the size of 256*256*3. One kind of the patches is the same as the input (labeled raw paired cross-domain image patches) for generator *G*; the other kind of patches is the labeled paired cross-domain image patches generated by generator *G* (these labels are the same as the input of generator *G*). The difference between the two input images is that one is the original image data and the other is the generated image data. These two kinds of image data enrich the input of discriminator *D*, so that discriminator *D* would have stronger discriminating ability after training. The output of discriminator *D* is a binary value. If the input image patch pairs match, the output of discriminator *D* is 1; otherwise, the output of discriminator *D* is 0. In detail, discriminator *D* consists of a Siamese network with a metric network. LeNet [52] is used for the two branches, whose weights are not shared. Because the outputs of the two branches are feature maps, we stretch the feature maps into

two one-dimensional vectors (2304-dimensional); then, we concatenate the two vectors into one vector (4608-dimensional) and feed it into a metric network of two fully connected layers, whose output channels are 1024 and 1. The non-linear activate functions for the first fully connected layer and the last fully connected layer are ReLU and Sigmoid, respectively.

In summary, the training data for AE-GAN-Net are a set of cross-domain image patch pairs with associated labels, and the AE-GAN-Net outputs 128-dimensional robust feature descriptors learned by the encoder in generator G.

3.2. Loss Function

To learn invariant feature descriptors for cross-domain images, a domain-consistent loss, which consists of content, feature consistency, and adversarial losses, is proposed to optimize AE-GAN-Net. We follow the optimization approach in GAN training for this minimax two player game setting and consider the following optimization problem that characterizes the interplay between generator G and discriminator D:

$$\begin{aligned}\min_{\theta}\max_{\beta} L\left(G_\theta, D_\beta\right) =& \mathbb{E}_{(C,R)\sim p(C,R)}\left[\log D_\beta(C,R)\right] \\ &+ \mathbb{E}_{C\sim p(C),R\sim p(R)}\left[\log\left(1-D_\beta\left(G_\theta(C),G_\theta(R)\right)\right)\right] \\ &+ \lambda\left(L_{content}+L_{feature}\right)\end{aligned} \tag{1}$$

where $\mathbb{E}$ is the Expectation; (C,R) are the inputs of labeled raw paired image patches; C is the ground camera image patches; R is the rendered image patches; $p(C,R)$ is a distribution over (C,R), $p(C)$ is a distribution over C and $p(R)$ is a distribution over R; G and D are the generator and discriminator, respectively; θ and β are the parameters of G and D, respectively; $\lambda > 0$ is a trade-off constant; $L_{content}$ and $L_{feature}$ are the content loss and feature consistency loss which are defined by Equations (4) and (5), respectively.

In detail, to learn a generator distribution $p(C,R)$ over data (C,R), the generator G_θ builds a mapping function from the prior noise distribution $p(C)$ and $p(R)$ to the data space as $G_\theta(C)$ and $G_\theta(R)$, and the discriminator, D_β, outputs a single scalar representing the probability that (C,R) came from training data rather than $p(C,R)$. G and D are both trained simultaneously: we adjust parameters θ for G to minimize $\log\left(1-D_\beta\left(G_\theta(C),G_\theta(R)\right)\right)$, and adjust parameters β for D to minimize $\log D_\beta(C,R)$, as if they are following the two-player min-max game with value function $L\left(G_\theta, D_\beta\right)$.

Content loss. To extract the common features for the matching image patch pairs of ground camera images and rendered images, the pixel-wise Mean Squared Error (MSE) loss is used to minimize the two AutoEncoders with the shared decoder for the input of C and R:

$$L_{C-content} = \frac{1}{NWH}\sum_{n=1}^{N}\sum_{x=1}^{W}\sum_{y=1}^{H}\left(C_{n,x,y}-(G_\theta(C))_{n,x,y}\right)^2 \tag{2}$$

$$L_{R-content} = \frac{1}{NWH}\sum_{n=1}^{N}\sum_{x=1}^{W}\sum_{y=1}^{H}\left(R_{n,x,y}-(G_\theta(R))_{n,x,y}\right)^2 \tag{3}$$

where the size of the fed image patches is $W \times H$, N is the channel of the image, and $G_\theta(C)$ and $G_\theta(R)$ are image patches generated by shared decoder in generator G. Thus, the total content loss is

$$L_{content} = L_{C-content} + L_{R-content} \tag{4}$$

Feature consistency loss. Our goal is to learn invariant feature descriptors for the cross-domain images. Thus, we use the following intuitive margin-based contrastive loss to constrain the feature

descriptors of the labeled image patch pairs (To reflect patch similarity, Euclidean distance is used for the feature descriptors):

$$L_{feature} = \frac{1}{2} l D_f^2 + \frac{1}{2}(1-l)\{max(0, m - D_f)\}^2 \tag{5}$$

where l is the label of the paired cross-domain image patches (if matched, $l = 1$; otherwise, $l = 0$); $D_f = \|f_C - f_R\|$ is the Euclidean distance between feature descriptors f_C and f_R (learned by the encoders in generator G) of C and R, respectively. Such feature consistency loss encourages the feature descriptors of matching pairs to be close and non-matching pairs to be separated by a distance of at least a margin m.

Specifically, non-matching cross-domain image patch pairs contribute to the margin-based contrastive loss only if their feature distance is smaller than the margin m. Equation (5) encourages the matching cross-domain image patches to be close in the feature space, and punishes the non-matching cross-domain image patches that are margin m away. As seen from the second part of Equation (5), the non-matching pairs with the feature distance larger than margin m will not contribute to margin-based contrastive loss. In fact, if the margin m is set too small, the feature consistency loss will be optimized only over the set of matching pairs; on the contrary, a larger margin m hampers learning. In our experiment, we set margin m to 0.01.

Adversarial loss. We describe the adversarial loss with the training strategy of the generator and the discriminator. Details are as follows:

Discriminator D aims to discriminate correctly the input paired cross-domain image patches, whether or not they match. It should be noted that the input paired image patches contain labeled raw pairs of cross-domain image patches (inputs of AE-GAN-Net) and the labeled pairs of image patches generated by generator G (the labels are the same to the inputs of AE-GAN-Net). These two kinds of paired cross-domain image patches not only enable the discriminator D to discriminate raw inputs, but also be valid for the image patch pairs generated by generator G. Essentially, this is a data augmentation mode, which enhances discriminator D with stronger discriminating capability.

Specifically, we minimize the Binary Cross Entropy (BCE) [53] between the decisions of discriminator D and the label (matching or non-matching). Concretely, the loss functions for discriminator D with raw paired image patches and generated paired image patches are defined as follows:

$$L_{D_{raw}} = L_{BCE}(D_\beta(C, R), 1) + L_{BCE}(D_\beta(G_\theta(C, R)), 0) \tag{6}$$

$$L_{D_{gencrate}} = L_{BCE}\left(D_\beta\left(G_\theta(C), G_\theta(R)\right), 1\right) + L_{BCE}\left(D_\beta\left(G_\theta(C), G_\theta(R)\right), 0\right) \tag{7}$$

where label 1 denotes that the paired image patches (C, R) is a match; label 0 denotes the paired image patches (C, R) that are not a match. Thus, the loss function for discriminator D is

$$L_D = kL_{D_{raw}} + (1-k)L_{D_{generate}} \tag{8}$$

where $k = 1$ denotes the inputs that are raw paired cross-domain image patches; and $k = 0$ denotes the inputs that are paired image patches generated by generator G.

Generator G tries to minimize the Binary Cross Entropy (BCE) loss between the decision made by discriminator D and generate more realistic matching or non-matching paired image patches so that discriminator D becomes completely confused. Finally, the total loss used for training generator G can be defined as the weighted sum of all the terms, as follows:

$$\begin{aligned} L_G =& \lambda_1 \left(L_{BCE}\left(D_\beta\left(G_\theta(C), G_\theta(R)\right), 1\right) + L_{BCE}\left(D_\beta\left(G_\theta(C), G_\theta(R)\right), 0\right)\right) \\ &+ \lambda_2 L_{fea\ ture} + \lambda_3 \left(L_{C-content} + L_{R-content}\right) \end{aligned} \tag{9}$$

where λ_1, λ_2 and λ_3 are the weights of BCE, feature and content losses, respectively.

The training strategy is performed in an alternating fashion. First, discriminator D is updated by taking a mini-batch of labeled raw paired cross-domain image patches and a mini-batch of generated paired image patches (the outputs of generator G). Second, generator G is updated by using the same mini-batch of labeled raw paired cross-domain image patches.

In summary, after AE-GAN-Net has been trained to converge, the paired image patches, which are generated by generator G from the matching raw paired cross-domain image patches, are still similar. For example, regarding the testing of paired matching cross-domain image patches (c, r), which are fed into a trained AE-GAN-Net, we obtain $G_\theta(c) \approx G_\theta(r)$. Then, using the shared decoder, the learned feature descriptors of c and r are $f_c \approx f_r$, demonstrating that the matching cross-domain image feature descriptors learned by AE-GAN-Net are invariant.

3.3. Training Strategy

We implemented AE-GAN-Net with PyTorch framework, and trained it with a Nvidia 2080 Ti GPU. AE-GAN-Net is trained with labeled paired image patches of ground camera images and rendered images, and the generator and discriminator are minimax in an alternating fashion. All weights in both the generator and the discriminator are initialized by Gaussian distribution of zero-mean and standard deviation of 0.05. Batch size is set at 20. Both discriminator and generator are trained with the Adam optimizer [54], which is an extension to stochastic gradient descent. Initially, the learning rates for the generator and discriminator are set at 0.0001 and 0.0002, respectively, and they both decreased 5% after 3 epochs. The discriminator is optimized three times more frequently than the generators.

4. Experiments

4.1. Dataset

The cross-domain images adopted in this paper are the ground camera images and the corresponding rendered images, which were obtained from the Xiangan campus of Xiamen University, China, about 3-square-kilometer with 100+ buildings; 30,000+ vertical and oblique aerial images of the campus were captured by the UAVs. The 3D image-based point cloud was generated by the SfM algorithm, as shown in Figure 5. The resolution of this 3D image-based point cloud is about 2 cm. Then, we captured 10,000+ ground camera images by mobile phone (HUAWEI Mate 9 with Leica Dual Camera with a 12 MP RGB sensor and 20 MP monochrome sensor). During our tests, the GNSS error of HUAWEI Mate 9 mobile phone we used in the open outdoor is about 2 to 5 m. To acquire a camera pose, we set up the mobile phone on a handheld gimbal, which reduces the camera jitter, to capture images. In this way, we obtain synthetic images rendered from the 3D image-based point cloud, which are similar to the viewpoints of the corresponding ground camera images. Several samples of corresponding ground camera and rendered images are shown in Figure 6.

The training data and the testing data, which do not intersect, are from different buildings. We use the paired cross-domain images of 90+ buildings to created the training data, and use the paired cross-domain images of 10 buildings to create the testing data. In addition, both the training and testing data are collected from different days when the weather is either sunny or cloudy. The ground camera images of the training data are captured by HUAWEI Mate 9; the ground camera images of testing data are captured from both HUAWEI Mata 9 and IPHONE 7 Plus.

To acquire the labeled samples, a semi-automated software was developed to select the corresponding points between each pair of the ground camera and the rendered images. Assuming that the transformation between these two corresponding cross-domain images is a perspective transformations; then, by randomly collecting several patches in the rendered images, the matching patches in the corresponding ground camera images are subsequently obtained. In detail, there are two parts for the semi-automated software. One is done manually; the other is done automatically. The part done manually selects at least 4 pairs of corresponding points on the ground camera image and rendered image to calculate the perspective transformation. The part done automatically randomly

collects several rendered image patches in a rendered image, and then automatically maps these rendered image patches to the corresponding ground camera image through the above calculated perspective transformation to obtain the matching ground camera image patches.

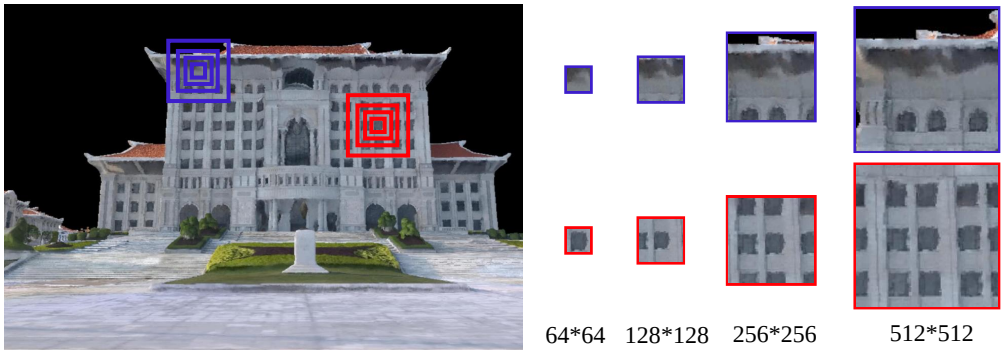
Figure 5. The generated 3D image-based point cloud of the Xiangan campus of Xiamen University.

Figure 6. The corresponding image pairs of ground camera images and rendered images. Top: Ground camera images; Bottom: Rendered images.

In addition, the size of the image patches must be carefully designed. If the size of a selected image patch is too small, the detailed information in the image patch will be limited; whereas, a large size patch becomes nearly a whole image, rather than a small image patch (Examples are illustrated in Figure 7). Image patches have a high probability of being selected in the blurred regions of the rendered images (e.g., the position of the blue bounding boxes in Figure 7a). It can be observed that if the size of the image patches is too small (e.g., 64*64 or 128*128), the patches will be in a completely blurred region, and the information loss will be very large (first row in Figure 7b). Thus, small blurred image patches are worthless. Additionally, buildings with many very similar repeating structures, such as the windows shown within the red bounding boxes in Figure 7a, lead to mismatches (second row in Figure 7b). In practical applications, we should consider the spatial resolution. However, to create a large amount of training data (paired matching ground camera image patches and rendered image patches) more conveniently, we did not capture the buildings very close to the camera, as shown by the corresponding image pairs in Figure 6. The reason is that if the distance is too short, buildings in the image will be large, and the size of the image patches cannot be well controlled. Therefore, the size of the image patches that we collected is between 256*256 and 512*512 pixels.

Based on the above considerations, 200,000+ matching and 200,000+ non-matching cross-domain image patch pairs were collected from the above 10,000+ corresponding ground camera and rendered images. The detailed distribution information of the 200,000+ matching cross-domain image patch

pairs is describe as follows: First, all the synthetic images rendered from the 3D image-based point clouds are subject to huge distortions. Thus, in the training data, the 200,000+ rendered images are also distorted. Second, as described in the third paragraph of Section 4.1, we set the size of the collected image patches to between 256 and 512 pixels. Thus, the number of image patches per size is about 700+ pairs. Third, because the position and orientation information obtained from the ground camera image are coarse, there is rotation bias between the ground camera images and corresponding rendered images. In addition, we only select the image pairs with a rotation bias of no more than 20 degrees, and vice versa. Several samples of collected cross-domain image patch pairs are shown in Figure 8. These labeled cross-domain image patches are the training data for AE-GAN-Net and comparative neural networks.



(**a**) (**b**)

Figure 7. Various size of rendered image patches in two typical locations. (**a**) Different bounding boxes of two locations in a rendered image. (**b**) Various size image patches of the bounding boxes in (**a**).

(**a**)

(**b**)

Figure 8. Several samples of collected cross-domain image patch pairs. The first row are the ground camera image patches, the second row are the rendered image patches. (**a**) Matching patch pairs of ground cameara images and rendered images. (**b**) Non-matching patch pairs of ground cameara images and rendered images.

4.2. Comparative Experiments

To demonstrate the superiority of AE-GAN-Net, we compared it with the existing mainstream feature descriptor learning neural networks. The TOP1 and TOP5 retrieval accuracy of the learned feature descriptors are used to measure performance. TOP1 retrieval is considered successful if the truly matched rendered image patch is ranked No.1. If the truly matched rendered image patch is retrieved in the first five ranked results, the TOP5 retrieval is considered successful. Results are listed in Table 1. To ensure the fairness of the comparisons, 6000+ pairs of matching cross-domain image patches were additionally collected as a retrieval benchmark dataset. Specifically, the data in the retrieval benchmark dataset are not used in the training data. The distribution information of testing set is the same as that in training set, which is described in the above paragraph.

Table 1. The TOP1 and TOP5 retrieval accuracy of proposed AE-GAN-Net and comparative networks on the cross-domain image patch retrieval dataset.

	AE-GAN-Net	H-Net++ [25]	DeepDesc [22]	Siam_l2 [44]	L2-Net [23]
TOP1	0.9028	0.7148	0.5358	0.3683	0.4608
TOP5	0.9335	0.8572	0.6672	0.4273	0.5015
	DeepCD [24]	TNet [27]	DOAP [28]	DDSAT [29]	DescNet [30]
TOP1	0.5535	0.5998	0.6338	0.6085	0.6118
TOP5	0.6427	0.6595	0.6863	0.6787	0.6906

As shown in Table 1, AE-GAN-Net achieves state-of-the-art retrieval performance for the learned feature descriptors on the established cross-domain image patch retrieval benchmark dataset, and shows significant improvement compared with the competing networks. DeepDesc [22] and Siam_l2 [44] are Siamese networks whose loss functions are constrained by Euclidean distance. DeepCD [24] is an asymmetric Siamese network. To help optimize loss function, L2-Net [23] introduces some constraints in the data sampling strategy. H-Net++ [25] incorporates the AutoEncoder into the Siamese network. TNet [27], DDSAT [29] and DescNet [30] use triplet loss with different data sampling strategies to learn the feature descriptors. DOAP [28] extracts local feature descriptors optimized by average precision. It can be viewed that whether or not the feature descriptors are learned by simple Siamese networks, variant Siamese networks, or triplet networks, the learned feature descriptors are not robust for retrieval on ground camera images and rendered images.

Moreover, we create another testing data, which are not used in the training data, to perform an accuracy assessment for our AE-GAN-Net and competing networks. The testing data is collected according to the following strategy: (1) Selecting 500 ground camera image patches, denoted as $C^+_{I_k}$, $k = 1, 2, 3 \ldots 500$; (2) Selecting 500 rendered image patches, which are matched to $C^+_{I_k}$, denoted as $R^+_{I_k}$, $k = 1, 2, 3 \ldots 500$; (3) Selecting 500 rendered image patches, which are non-matched to $C^+_{I_k}$, denoted as $R^-_{I_k}$, $k = 1, 2, 3 \ldots 500$. Thus, the testing data has 500 ground camera image patches and 1000 rendered image patches, which consists of 500 paired matching cross-domain image patches and 500 paired non-matching cross-domain image patches.

Then, using the above collected testing data, the detailed testing strategy is as follows: first, using the trained AE-GAN-Net to compute the feature descriptors of the above collected image patches; second, for each ground camera image patch in $C^+_{I_k}$, retrieving each ground camera image patch in $C^+_{I_k}$ in the $R^+_{I_k}$ and $R^-_{I_k}$ to obtain the TOP1 retrieval result. Third, we draw the ROC curve based on the precision and recall, which is calculated by the TOP1 retrieval result. The ROC curves are shown in Figure 9. From the ROC curves, it can be observed that our AE-GAN-Net has the best performance.

In addition, we also explored some ablation studies. First, to demonstrate the importance of the embedded GAN module, we removed GAN from AE-GAN-Net. However, it should be noted that without adversarial loss (GAN loss), AE-GAN-Net degenerates into a specific form of H-Net++ [25],

i.e., H-Net++ with a shared decoder. Results in Table 1 show that AE-GAN-Net is 18.8% better than H-Net++. Second, for the weight of the adversarial loss (Equation (9)), a small weight will lead to an inefficient constraint; whereas, a large weight will lead to mode-collapse. According to our experiments, $\lambda_1 = 0.1$, $\lambda_2 = 1.5$ and $\lambda_3 = 1$ (Equation (9)) are the most suitable weights for adversarial loss.

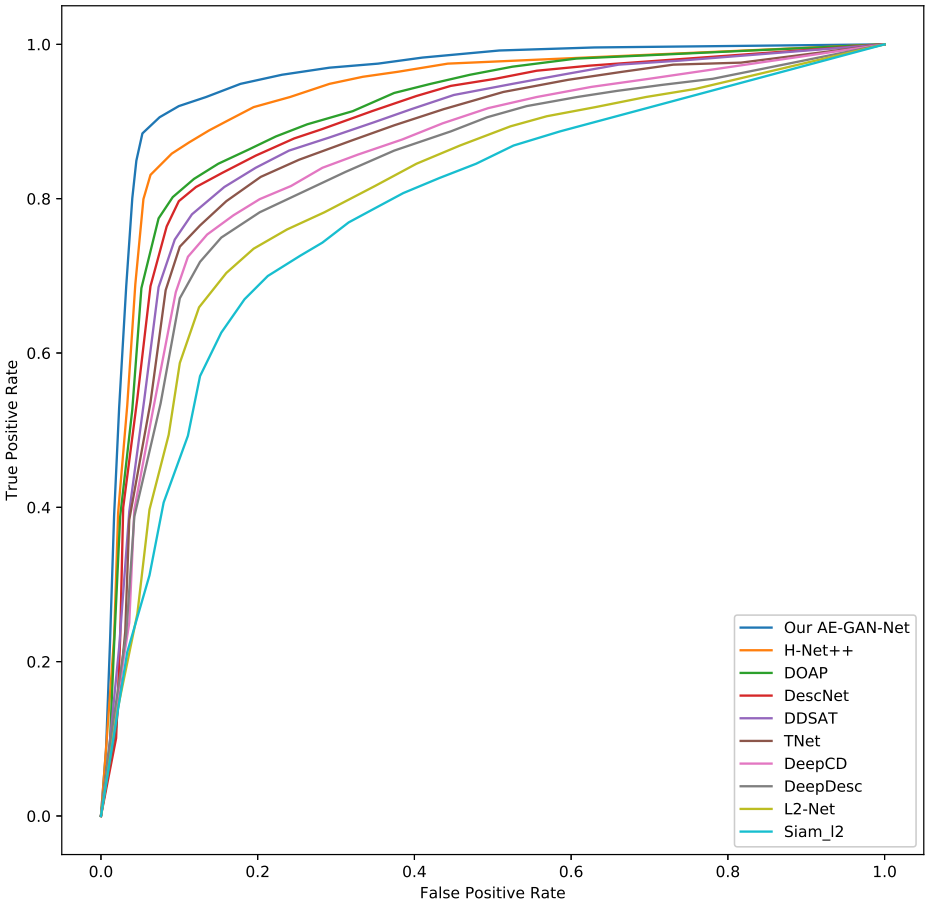


Figure 9. ROC curve based on testing data for our AE-GAN-Net and competing networks.

4.3. Invariance of the Learned Feature Descriptors

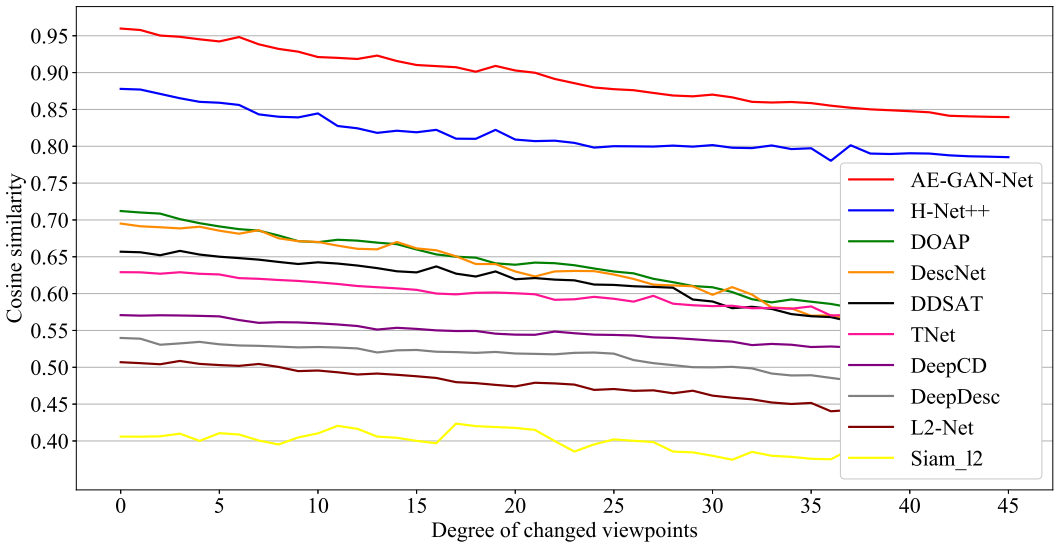
To demonstrate that the feature descriptors learned by AE-GAN-Net are invariant against distortion, viewpoint, spatial resolution, rotation and scaling, several experiments were performed with additionally collected datasets, as follows:

Distortion invariance. As described in Section 1, the synthetic images rendered from 3D image-based point clouds are usually distorted and of low resolution. In contrast, the details of the ground camera images are fine. To extract the consistent feature descriptors for these two extremely different cross-domain images, we propose the AE-GAN-Net to learn invariant features for the ground camera images and rendered images. The experimental results in Table 1 and Figures 9 show that the feature descriptors learned by AE-GAN-Net are invariant. Thus, from these experimental results, we conclude that AE-GAN-Net can overcome the distortion of the rendered image to extract feature descriptors with distortion invariance, i.e., the feature descriptors of the cross-domain image patches learned by AE-GAN-Net are invariant against distortion.

Viewpoint invariance. For the viewpoint invariance of the learned feature descriptors, we investigated cosine similarity for the learned feature descriptors on image patch pairs extracted from ground camera images and rendered images. First, we took a fixed viewpoint ground camera

image patch (Figure 10a) and the rendered image patches with different viewpoints from zero to 45 degrees (Figure 10b). Second, we calculated the cosine similarity for the learned feature descriptor vectors from the above collected paired cross-domain image patches. In detail, the two vectors used to calculate the cosine similarity are the paired feature descriptors of the paired ground camera image patches and rendered image patches. The paired feature descriptor vectors are computed by the trained AE-GAN-Net, i.e., the two purple vectors in Figure 4. Compared with other competing networks, the features extracted by AE-GAN-Net have the highest cosine similarity (the red line in Figure 10c). Thus, the cosine similarity measurement verifies that the cross-domain image feature descriptors learned by AE-GAN-Net are invariant against viewpoint variations. In addition, the training data for the network training contains only paired cross-domain image patches at a similar viewpoint, and the paired cross-domain image patches with largely deviated viewpoints are not included in the training data. Therefore, all the curves in Figure 10c show a decreasing trend.

(**a**) (**b**)



(**c**)

Figure 10. The cosine similarity of varied image patch pairs with different viewpoints. (**a**) Fixed patch. (**b**) Gradually varied rendered image patches with different viewpoints. (**c**) The cosine similarity of (**a**) and each patch in (**b**).

Spatial resolution and scale invariance. For the spatial resolution and scale invariance of the learned feature descriptors, we collected a multi-scale cross-domain paired patches dataset. We fixed a ground camera image patch with 300*300 pixels. Then, from the same viewpoint, we selected different sizes of rendered image patches (e.g., 251*251, 252*252, ..., 500*500 pixels) for matching. The feature cosine similarity of these multi-size cross-domain patches were still maintained at a high level (between 0.8231 and 0.9508). The reason is that the size of the collected training data is between 256*256 and 512*512 pixels. Thus, AE-GAN-Net has learned the feature of various sizes of cross-domain image

patches. This justifies that the features learned by AE-GAN-Net are invariant against spatial resolution and scale variations. An example of the cosine similarity of multi-size cross-domain patches is shown in Figure 11.

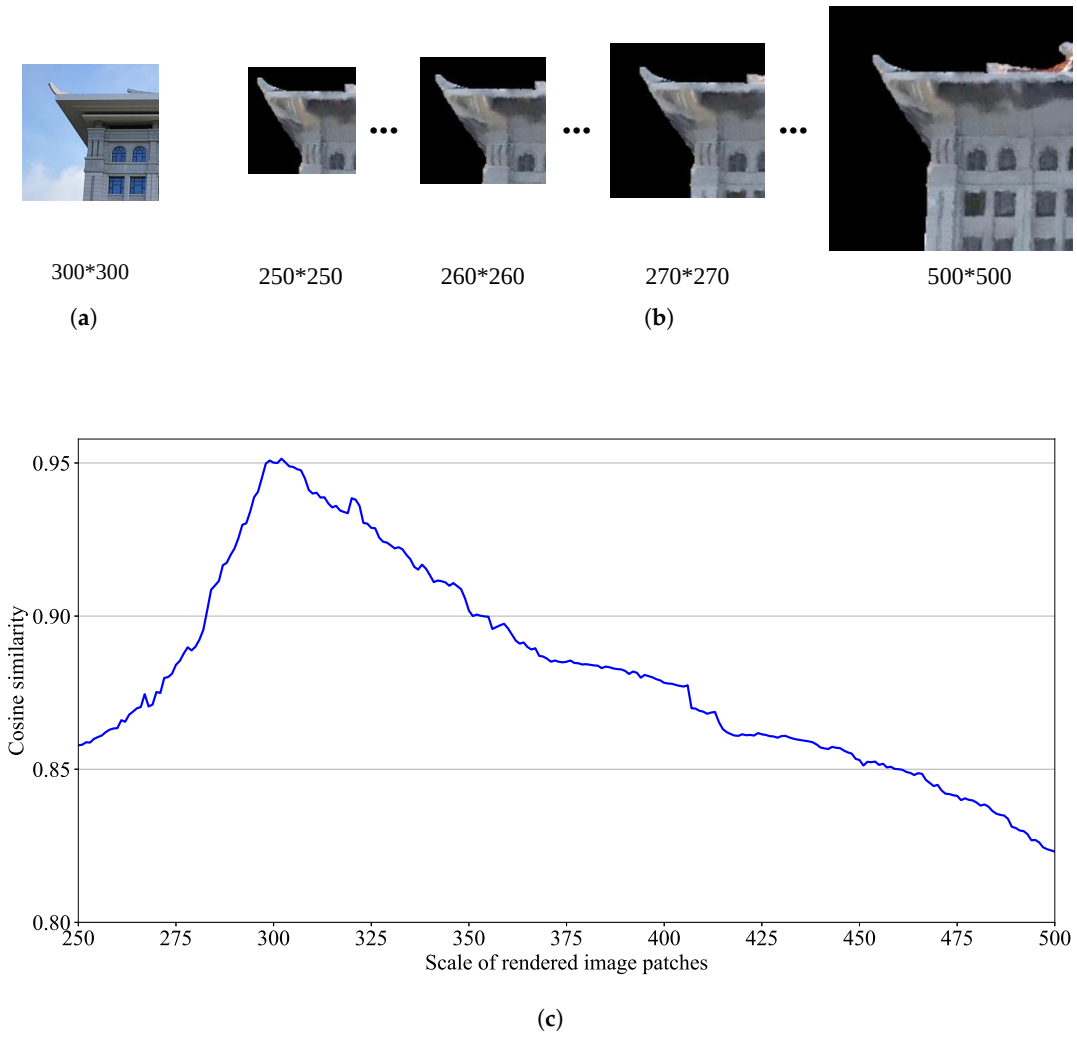
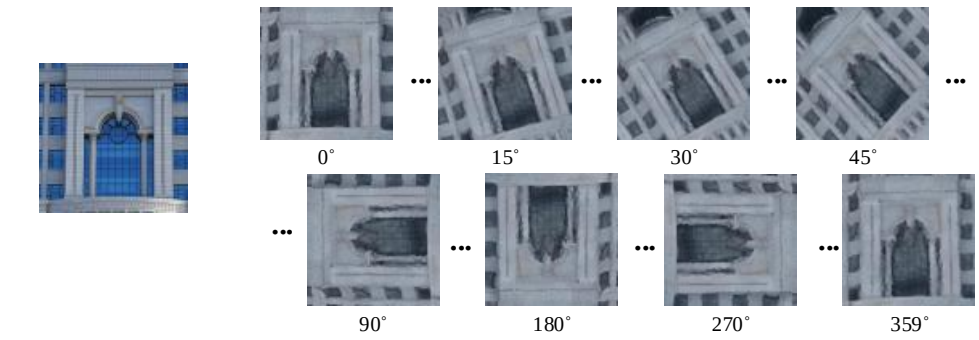
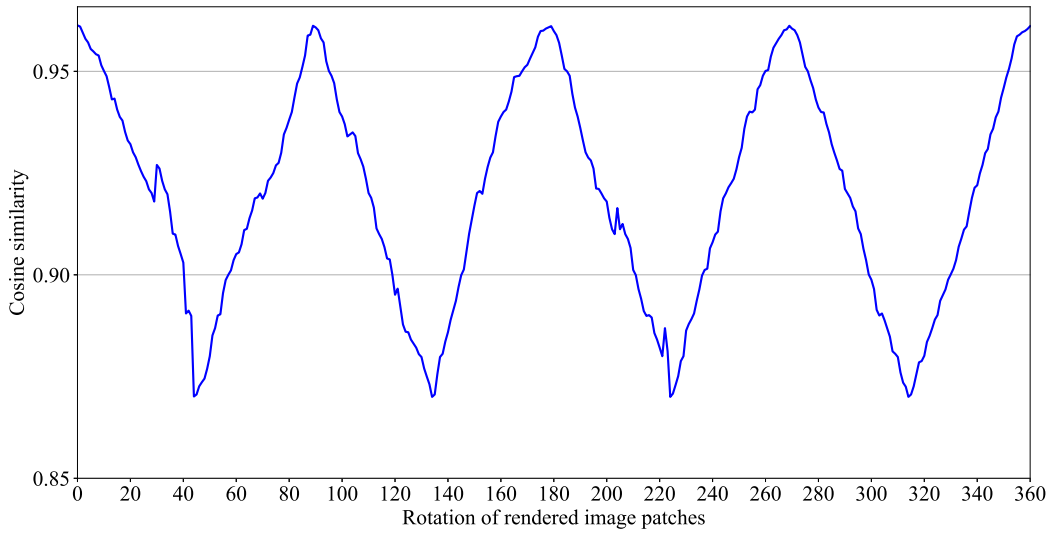


Figure 11. The cosine similarity of varied image patch pairs with different scale. (**a**) Fixed patch. (**b**) Gradually varied rendered image patches with different scale. (**c**) The cosine similarity of (**a**) and each patch in (**b**).

Rotation invariance. To justify rotation invariance of the learned feature descriptors, we collected a rotational cross-domain paired patches dataset. We fixed a ground camera image patch and then obtained a total of 360 rendered patches by rotating the rendered images at different angles (degree by degree, counterclockwise selecting from the corresponding positions of the fixed ground camera image patch). The feature cosine similarity of these rotational cross-domain patches is still maintained at a high level (between 0.8701 and 0.9613). There is an inherent bias between the collected corresponding cross-domain images caused by the positioning error, including the viewpoints and positional bias, which cause the corresponding cross-domain images to have rotational deviation. Thus, the training data also contains the cross-domain image patch pairs with rotation offset. An example of the cosine similarity of rotational cross-domain patches is shown in Figure 12.



(**a**) (**b**)



(**c**)

Figure 12. The cosine similarity of varied image patch pairs with different rotation. (**a**) Fixed patch. (**b**) Gradually varied rendered image patches with different rotation. (**c**) The cosine similarity of (**a**) and each patch in (**b**).

In addition, to better intuitively demonstrate the invariance of the feature descriptors learned by proposed AE-GAN-Net, we visualize the feature histogram of the learned feature descriptors in Figure 13. The content shown in the histogram is the 128-dimensional feature descriptor vector learned by AE-GAN-Net. In the histogram, the x-axis is the dimension of the feature vector, and the y-axis is the value of the learned 128-dimensional feature descriptor vector. It can be observed that the feature distribution of each pair of matching cross-domain image patches is consistent, and the values of each dimension are similar. Thus, from the visualization of the similarity of the features, we conclude that the feature descriptors learned by AE-GAN-Net are invariant.

Note that the embedded GAN module in AE-GAN-Net is used to determine whether the cross-domain image patch pairs generated by the generator are similar. After training with a huge number of cross-domain image patch pairs, the strong performance of GAN in AE-GAN-Net enables it to decide if a pair of image patch pairs is a match or not. These image patch pairs may have distortions, various sizes, and rotation offsets. Thus, combined with the shared decoder and the constraint of feature loss, the feature descriptors learned by AE-GAN-Net are invariant.

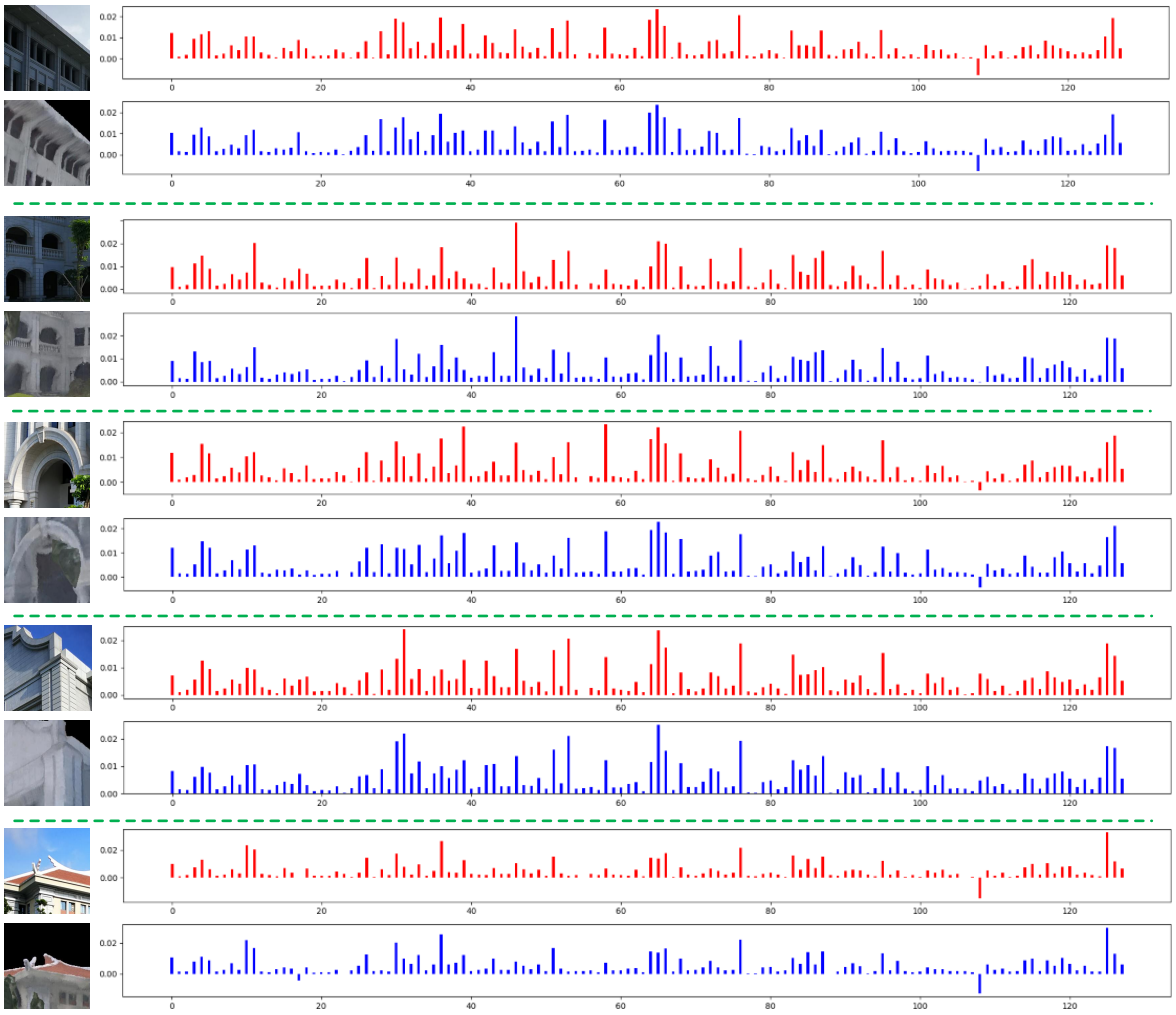
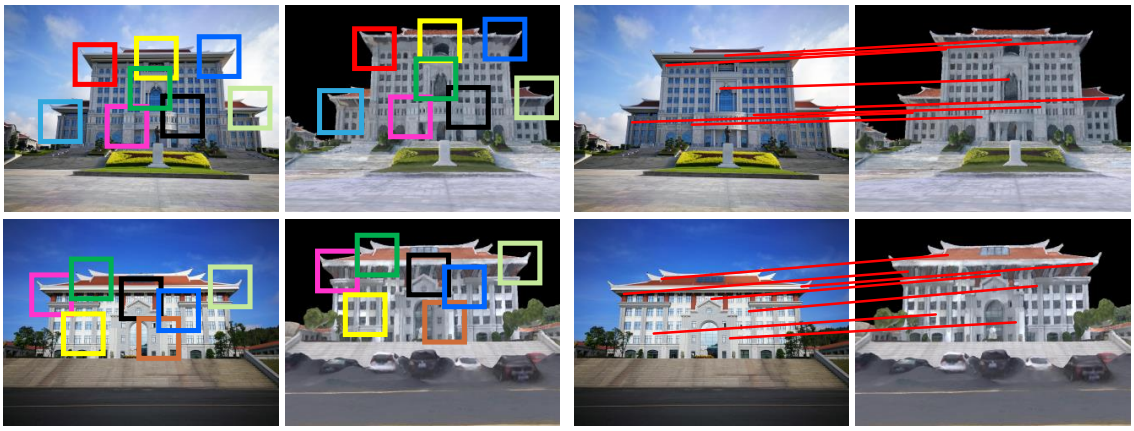


Figure 13. The histogram visualization feature descriptors learned by AE-GAN-Net on the matching patch paris of ground camera images and rendered images. For each pair of matching cross-domain image patches, the top one is ground camera image patch, the bottom one is the matching rendered image patch.

4.4. Image Matching and AR Applications

To match the ground camera images and the rendered images, the following four-step strategy is adopted: (1) Randomly select 2,000 points in each paired cross-domain images, and select 3 patches at each point (250*250, 375*375, 500*500 pixels); (2) Use the trained AE-GAN-Net to compute feature descriptors for the above selected image patches; (3) Apply the Nearest Neighbor Search (NNS) algorithm for retrieval and retain only the TOP1 retrieved results with cosine similarities greater than 0.9; (4) Apply RANSAC to filter mismatched results. In Figure 14, we show the image patch matching results and center point connection of matching patches for the two pairs of cross-domain images in Figure 3.

As shown in Figure 15, the real-time library information, cartoon spider-man, NO PARKING sign and Welcome sign are registered into the open outdoor environments by the inferred spatial relationship. These AR applications demonstrate that the proposed virtual–real registration method is feasible for use in open and outdoor environments. However, the results have some limitations: the virtual objects have slight deformation after virtual–real registration, and there is a gap between virtual objects and targets (e.g., the gap between the cartoon spider-man and the building in Figure 15).



(**a**) (**b**)

Figure 14. Cross-domain image patch matching results and center point connection of matching image patches. Left: Ground camera images; Right: Rendered images with the same viewpoint of ground camera image. (**a**) Cross-domain image patch matching results. (**b**) Center point connection of the matching image patches in (**a**).

Figure 15. AR applications by the proposed virtual–real registration approach.

5. Discussion and Analysis

Advantage of network framework. The framework of AE-GAN-Net consists of an AutoEncoder and GAN. For extracting domain-specific information, the AutoEncoder is more effective than CNN-based Siamese and triplet networks. The embedded GAN is used to constrain the generated image pairs; if the raw inputs are matching paired cross-domain images, the generated image pairs are similar, and vice versa. Thus, the embedded GAN improves the preservation of information in the feature encoder section. With the shared decoder, this constraint mechanism enables AE-GAN-Net to learn essential domain-consistent feature descriptors for ground camera and rendered images in dynamic balancing.

Effectiveness of domain-consistent loss. The content loss is used to reconstruct the generated images similar to the input images; the feature consistency loss is an intuitive loss, which constrains the feature descriptors unified in Euclidean space. Importantly, the adversarial loss makes the parameters of AE-GAN-Net update in an alternating fashion. This mechanism preserves the image content representation and balance feature descriptors consistency for ground camera images and rendered images.

Benefit of the discriminator. The training data for the discriminator contains not only raw labeled paired cross-domain images, but also the image pairs generated by the generator. These two kinds of image data make the discriminator more discriminative. Therefore, if the inputs are matching cross-domain images, the stronger discriminator will help the generator generate more similar image pairs.

In addition, there is a problem regarding lens distortion. In fact, deep learning networks can eliminate the influence of lens distortion when learning the features from images. For example, ResNet [55] and Faster R-CNN [56] are trained with ImageNet [57] for image classification, recognition, and detection. The training data contains a large amount of image data from different cameras with different lens distortions. However, the above deep learning networks still learn robust features for

their tasks without being affected by the lens distortion. Thus, the lens distortion is eliminated by deep learning networks, including our proposed AE-GAN-Net.

Although feature descriptors learned by AE-GAN-Net for ground camera images and rendered images are invariant, there are still some limitations. If the images have large occlusion and distortion, the learned feature descriptors will be meaningless and will result in failed matching results. In fact, image matching for very poor-quality cross-domain images often fails and sometimes beyond human capabilities for such tasks. In addition, we only focus on verifying the feasibility of our proposed virtual–real registration of AR in outdoor environments. Not considering the impact of the spatial resolution of the images is the drawback of this paper. Currently, our method cannot deal with the close-up images. In future work, we will try to solve this problem.

Moreover, rendered images rely heavily on positioning information (position and orientation) from the mobile phone. Thus, our proposed virtual–real registration approach of AR in outdoor environments is efficient and intuitive, but it is limited to open outdoor environments. For a dense building environment, more sensors may be needed to obtain accurate positioning.

6. Conclusions

In this paper, we presented an end-to-end network, AE-GAN-Net, to learn the invariant features for ground camera images and synthetic images rendered from 3D image-based point clouds. The proposed AE-GAN-Net uses the embedded GAN to constrain the similarity of image pairs generated by the AutoEncoders. According to the proposed domain-consistent loss, which balances the image content and consistency of the feature descriptors for image pairs of ground camera images and rendered images, the feature descriptors learned by AE-GAN-Net are invariant against distortion, viewpoints, spatial resolution, rotation and scale variations. Extensive experimental results on our cross-domain image patch retrieval benchmark dataset show that AE-GAN-Net achieves state-of-the-art patch-based retrieval performance. Using the matching relationship of ground camera images and rendered images to indirectly infer the spatial relationship between 3D and 2D space, we explored and verified an intuitive solution for the virtual–real registration of AR in an open outdoor campus environment. To achieve more accurate virtual–real registration of AR in outdoor environments, we will explore more accurate patch matching for ground camera images and rendered images in future work. Additionally, we plan to extend the AE-GAN-Net framework to other cross-domain image problems.

Author Contributions: Conceptualization, W.L. (Weiquan Liu) and C.W.; Methodology, W.L. (Weiquan Liu); Software, W.L. (Weiquan Liu) and X.B.; Validation, W.L. (Weiquan Liu), X.B and X.L.; Formal Analysis, W.L. (Weiquan Liu), S.C. and W.L. (Wei Li); Investigation, W.L. (Weiquan Liu) and S.C.; Resources, W.L. (Weiquan Liu) and C.W.; Data Curation, W.L. (Weiquan Liu) and Y.L.; Writing—Original Draft Preparation, W.L. (Weiquan Liu); Writing—Review and Editing, W.L. (Weiquan Liu), S.L. and S.C.; Visualization, D.W.; Supervision, C.W. and J.L.; Project Administration, C.W. and J.L.; Funding Acquisition, C.W. and J.L.

Funding: This research was funded by National Natural Science Foundation of China, grant number U1605254, 41471379 and 61371144.

Acknowledgments: We thank the reviewers for their careful reading and valuable comments, which helped us to improve the manuscript.

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

References

1. Akçayır, M.; Akçayır, G. Advantages and challenges associated with augmented reality for education: A systematic review of the literature. *Educ. Res. Rev.* **2017**, *20*, 1–11. [CrossRef]
2. Rao, J.; Qiao, Y.; Ren, F.; Wang, J.; Du, Q. A mobile outdoor augmented reality method combining deep learning object detection and spatial relationships for geovisualization. *Sensors* **2017**, *17*, 1951. [CrossRef] [PubMed]
3. Portalés, C.; Lerma, J.L.; Navarro, S. Augmented reality and photogrammetry: A synergy to visualize physical and virtual city environments. *ISPRS J. Photogramm. Remote. Sens.* **2010**, *65*, 134–142. [CrossRef]

4. Luchetti, G.; Mancini, A.; Sturari, M.; Frontoni, E.; Zingaretti, P. Whistland: An augmented reality crowd-mapping system for civil protection and emergency management. *ISPRS Int. J. Geo-Inf.* **2017**, *6*, 41. [CrossRef]
5. Huang, W.; Sun, M.; Li, S. A 3D GIS-based interactive registration mechanism for outdoor augmented reality system. *Expert Syst. Appl.* **2016**, *55*, 48–58. [CrossRef]
6. Pellas, N.; Fotaris, P.; Kazanidis, I.; Wells, D. Augmenting the learning experience in primary and secondary school education: A systematic review of recent trends in augmented reality game-based learning. In *Virtual Reality*; Springer: Cham, Switzerland, 2018; pp. 1–18.
7. Chen, P.; Liu, X.; Cheng, W.; Huang, R. A review of using Augmented Reality in Education from 2011 to 2016. In *Innovations in Smart Learning*; Springer: Cham, Switzerland, 2017; pp. 13–18.
8. Bernhardt, S.; Nicolau, S.A.; Soler, L.; Doignon, C. The status of augmented reality in laparoscopic surgery as of 2016. *Med. Image Anal.* **2017**, *37*, 66–90. [CrossRef] [PubMed]
9. Lan, L.; Xia, Y.; Li, R.; Liu, K.; Mai, J.; Medley, J.A.; Obeng-Gyasi, S.; Han, L.K.; Wang, P.; Cheng, J.X. A fiber optoacoustic guide with augmented reality for precision breast-conserving surgery. *Light Sci. Appl.* **2018**, *7*, 2. [CrossRef] [PubMed]
10. Pelargos, P.E.; Nagasawa, D.T.; Lagman, C.; Tenn, S.; Demos, J.V.; Lee, S.J.; Bui, T.T.; Barnette, N.E.; Bhatt, N.S.; Ung, N.; et al. Utilizing virtual and augmented reality for educational and clinical enhancements in neurosurgery. *J. Clin. Neurosci.* **2017**, *35*, 1–4. [CrossRef] [PubMed]
11. Pang, Y.; Yuan, M.; Nee, A.Y.; Ong, S.K.; Youcef-Toumi, K. A markerless registration method for augmented reality based on affine properties. In Proceedings of the 7th Australasian User Interface Conference—Volume 50, Hobart, Australia, 16–19 January 2006; pp. 25–32.
12. Yuan, M.; Ong, S.K.; Nee, A.Y. A generalized registration method for augmented reality systems. *Comput. Graph.* **2005**, *29*, 980–997. [CrossRef]
13. Panou, C.; Ragia, L.; Dimelli, D.; Mania, K. An Architecture for Mobile Outdoors Augmented Reality for Cultural Heritage. *ISPRS Int. J. Geo-Inf.* **2018**, *7*, 463. [CrossRef]
14. Azuma, R.; Baillot, Y.; Behringer, R.; Feiner, S.; Julier, S.; MacIntyre, B. Recent advances in augmented reality. *IEEE Comput. Graph. Appl.* **2001**, *21*, 34–47. [CrossRef]
15. Schonberger, J.L.; Frahm, J.M. Structure-from-motion revisited. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016, pp. 4104–4113.
16. Jensen, J.; Mathews, A. Assessment of image-based point cloud products to generate a bare earth surface and estimate canopy heights in a woodland ecosystem. *Remote. Sens.* **2016**, *8*, 50. [CrossRef]
17. Hung, C.; Xu, Z.; Sukkarieh, S. Feature learning based approach for weed classification using high resolution aerial images from a digital camera mounted on a UAV. *Remote Sens.* **2014**, *6*, 12037–12054. [CrossRef]
18. Lowe, D.G. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. [CrossRef]
19. Bay, H.; Tuytelaars, T.; Van Gool, L. Surf: Speeded up robust features. In Proceedings of the European Conference on Computer Vision, Graz, Austria, 7–13 May 2006; pp. 404–417.
20. Tola, E.; Lepetit, V.; Fua, P. Daisy: An efficient dense descriptor applied to wide-baseline stereo. *IEEE Trans. Pattern Anal. Mach. Intell.* **2009**, *32*, 815–830. [CrossRef] [PubMed]
21. Rublee, E.; Rabaud, V.; Konolige, K.; Bradski, G.R. ORB: An efficient alternative to SIFT or SURF. In Proceedings of the IEEE International Conference on Computer Vision. Citeseer, Barcelona, Spain, 6–13 November 2011; Volume 11, p. 2.
22. Simo-Serra, E.; Trulls, E.; Ferraz, L.; Kokkinos, I.; Fua, P.; Moreno-Noguer, F. Discriminative learning of deep convolutional feature point descriptors. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 118–126.
23. Tian, Y.; Fan, B.; Wu, F. L2-net: Deep learning of discriminative patch descriptor in euclidean space. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 661–669.
24. Yang, T.Y.; Hsu, J.H.; Lin, Y.Y.; Chuang, Y.Y. Deepcd: Learning deep complementary descriptors for patch representations. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 3314–3322.

25. Liu, W.; Shen, X.; Wang, C.; Zhang, Z.; Wen, C.; Li, J. H-Net: Neural Network for Cross-domain Image Patch Matching. In Proceedings of the International Joint Conferences on Artificial Intelligence (IJCAI), Stockholm, Sweden, 13–19 July 2018; pp. 856–863.
26. Schroff, F.; Kalenichenko, D.; Philbin, J. Facenet: A unified embedding for face recognition and clustering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 815–823.
27. Kumar, B.; Carneiro, G.; Reid, I. Learning local image descriptors with deep siamese and triplet convolutional networks by minimising global loss functions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA , 27–30 June 2016; pp. 5385–5394.
28. He, K.; Lu, Y.; Sclaroff, S. Local descriptors optimized for average precision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 596–605.
29. Keller, M.; Chen, Z.; Maffra, F.; Schmuck, P.; Chli, M. Learning deep descriptors with scale-aware triplet networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 2762–2770.
30. Dong, Y.; Jiao, W.; Long, T.; Liu, L.; He, G.; Gong, C.; Guo, Y. Local Deep Descriptor for Remote Sensing Image Feature Matching. *Remote Sens.* **2019**, *11*, 430. [CrossRef]
31. Lenc, K.; Vedaldi, A. Learning covariant feature detectors. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; pp. 100–117.
32. Charte, D.; Charte, F.; García, S.; del Jesus, M.J.; Herrera, F. A practical tutorial on autoencoders for nonlinear feature fusion: Taxonomy, models, software and guidelines. *Inf. Fusion* **2018**, *44*, 78–96. [CrossRef]
33. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014; pp. 2672–2680.
34. Tayara, H.; Ham, W.; Chong, K. A real-time marker-based visual sensor based on a FPGA and a soft core processor. *Sensors* **2016**, *16*, 2139. [CrossRef]
35. Kawai, N.; Sato, T.; Nakashima, Y.; Yokoya, N. Augmented reality marker hiding with texture deformation. *IEEE Trans. Vis. Comput. Graph.* **2016**, *23*, 2288–2300. [CrossRef]
36. Bach, B.; Sicat, R.; Beyer, J.; Cordeil, M.; Pfister, H. The hologram in my hand: How effective is interactive exploration of 3D visualizations in immersive tangible augmented reality? *IEEE Trans. Vis. Comput. Graph.* **2017**, *24*, 457–467. [CrossRef]
37. Chen, J.; Cao, R.; Wang, Y. Sensor-aware recognition and tracking for wide-area augmented reality on mobile phones. *Sensors* **2015**, *15*, 31092–31107. [CrossRef] [PubMed]
38. Yi, K.M.; Trulls, E.; Lepetit, V.; Fua, P. Lift: Learned invariant feature transform. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; pp. 467–483.
39. Lin, T.Y.; Cui, Y.; Belongie, S.; Hays, J. Learning deep representations for ground-to-aerial geolocalization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 5007–5015.
40. Melekhov, I.; Kannala, J.; Rahtu, E. Image patch matching using convolutional descriptors with euclidean distance. In Proceedings of the Asian Conference on Computer Vision, Taipei, Taiwan, 20–24 November 2016; pp. 638–653.
41. Bromley, J.; Guyon, I.; LeCun, Y.; Säckinger, E.; Shah, R. Signature verification using a "siamese" time delay neural network. In Proceedings of the Advances in Neural Information Processing Systems, Denver, CO, USA, 28 November–1 December 1994; pp. 737–744.
42. Chopra, S.; Hadsell, R.; LeCun, Y. Learning a similarity metric discriminatively, with application to face verification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Diego, CA, USA, 20–26 June 2005; pp. 539–546.
43. Han, X.; Leung, T.; Jia, Y.; Sukthankar, R.; Berg, A.C. Matchnet: Unifying feature and metric learning for patch-based matching. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3279–3286.
44. Zagoruyko, S.; Komodakis, N. Learning to compare image patches via convolutional neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015, pp. 4353–4361.

45. Brown, M.; Hua, G.; Winder, S. Discriminative learning of local image descriptors. *IEEE Trans. Pattern Anal. Mach. Intell.* **2010**, *33*, 43–57. [CrossRef] [PubMed]
46. Krystian, M.; Schmid, C. A performance evaluation of local descriptors. *IEEE Trans. Pattern Anal. Mach. Intell.* **2005**, *27*, 1615–1630.
47. Balntas, V.; Lenc, K.; Vedaldi, A.; Mikolajczyk, K. HPatches: A benchmark and evaluation of handcrafted and learned local descriptors. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 5173–5182.
48. Hu, Y.; Gibson, E.; Vercauteren, T.; Ahmed, H.; Emberton, M.; Moore, C.; Noble, J.; Barratt, D. Intraoperative organ motion models with an ensemble of conditional generative adversarial networks. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Quebec City, QC, Canada, 11–13 September 2017; pp. 368–376.
49. Wang, T.C.; Liu, M.Y.; Zhu, J.Y.; Tao, A.; Kautz, J.; Catanzaro, B. High-resolution image synthesis and semantic manipulation with conditional gans. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8798–8807.
50. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Proceedings of the International Conference on Machine Learning, Lille, France, 6–11 July 2015; pp. 448–456.
51. Klambauer, G.; Unterthiner, T.; Mayr, A.; Hochreiter, S. Self-normalizing neural networks. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 971–980.
52. LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
53. Nie, D.; Trullo, R.; Lian, J.; Petitjean, C.; Ruan, S.; Wang, Q.; Shen, D. Medical image synthesis with context-aware generative adversarial networks. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Quebec City, QC, Canada, 11–13 September 2017; pp. 417–425.
54. Kingma, D.; Ba, J. Adam: A Method for Stochastic Optimization. *arXiv* **2014**, arXiv:1412.6980 .
55. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
56. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 7–12 December 2015; pp. 91–99.
57. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 248–255.