Geometry-Aware Satellite-to-Ground Image Synthesis for Urban Areas

Xiaohu Lu[1*] Zuoyue Li[2*] Zhaopeng Cui[2†] Martin R. Oswald[2]
Marc Pollefeys[2,3] Rongjun Qin[1†]
[1]The Ohio State University [2]ETH Zürich [3]Microsoft

Abstract

We present a novel method for generating panoramic street-view images which are geometrically consistent with a given satellite image. Different from existing approaches that completely rely on a deep learning architecture to generalize cross-view image distributions, our approach explicitly loops in the geometric configuration of the ground objects based on the satellite views, such that the produced ground view synthesis preserves the geometric shape and the semantics of the scene. In particular, we propose a neural network with a geo-transformation layer that turns predicted ground-height values from the satellite view to a ground view while retaining the physical satellite-to-ground relation. Our results show that the synthesized image retains well-articulated and authentic geometric shapes, as well as texture richness of the street-view in various scenarios. Both qualitative and quantitative results demonstrate that our method compares favorably to other state-of-the-art approaches that lack geometric consistency.

1. Introduction

Due to the increasing availability of satellite images (e.g. Google Earth) it is nowadays possible to cover almost every single corner of the world, yet such a capacity for ground-view images does not exist. Being able to generate a consistent ground view from a given satellite image is extremely useful for applications such as wide-area virtual model generation, media content enrichment, 3D realistic gaming, simulations and cross-view matching. This problem is known as a satellite-to-ground cross-view synthesis. In this work, we address this problem by proposing a geometry-aware framework that preserves the geometric and the relative geographical locations of the ground objects by fully utilizing information extracted from a satellite image. Our goal is to represent the ground-view as geometrically realistic as possible.

This raises several unique and difficult challenges: first of all, the view differences are drastic such that the information extracted from one view usable for inferring the other is highly limited. For instance, we may only observe the rooftop of a building in the satellite view with very little or no information about the facades. Secondly, the resolution of the inferable information from the satellite view is too coarse as compared to the ground images (normally the common regions might just be the ground), thus directly using partial information from the satellite view to generate ground views are difficult. Thirdly, the ground-view images generally exhibit much more local details than satellite views. As for example in an urban scenario, there exist many dynamic objects such as pedestrians and vehicles. Also, places with visibly similar street patterns in the satellite view might look completely different in the ground view, which can present a one-to-many mapping leading to the lack of diversity when the synthesized ground-view image is conditioned to the satellite views. Finally, due to the limited accuracy and availability of the GPS (Global Positioning System) information of images, the alignment between the satellite and ground-view is often insufficient to serve as training data for learning-based methods.

Satellite patch Predicted street-view panorama

Figure 1: **Two examples of our satellite-to-ground imaging synthesis.** Given a single satellite image patch, we learn to predict a corresponding street-view RGB panorama in an end-to-end fashion by leveraging geometric information.

Recently there are several works trying to solve simi-

*These authors contributed equally to this work.
†Corresponding authors.

lar problems. Zhai *et al*. [23] proposed a deep learning method to generate plausible ground-level panoramas from aerial images. Features are learned and extracted from the aerial image and the transformation to the ground level was formed through learning a per-pixel transformation, which is further used to generate RGB images through a generative model. Regmi *et al*. [15] proposed to learn the semantic segmentation together with RGB images within a uniform conditional generative adversarial network (GAN) architecture. Because there is no geometric transformation encoded in the network, the synthesized image may be quite different from the real one in geometry although it may look reasonable. So they further improved their method in [16] and use geometric constraints to add some details of roads in the generated image. However, only a simple homography for the overlapping road transformation was considered in this approach and the generation of other objects completely relied on learned transformations which leads to scenes with a large amount of hallucinated content.

In virtue of the widely available satellite images, we propose to address this problem by using the learned height and semantic representations from such a dataset to form a projective transformation between satellite and ground. This allows to fully utilize the geometric information represented in the satellite view for reality-based cross-view synthesis. We utilize this as a novel cross-view image transformation module to transform both semantic and color information from a satellite image to the street view. This is followed by a per-pixel generative model for generating plausible ground-view RGB images using information from the transformation module. As the transformation represents the actual projective relationship cross the two different views, our generated ground-view image is more geometrically meaningful, thus yielding more realistic panoramic image textures. Our experiments show that our method outperforms state-of-the-art methods.

Our **contributions** are as follows. Firstly, we propose an end-to-end network structure ensemble that exploits the geometric constraints of the street-view image generation from a satellite image. Secondly, we present a novel cross-view image transformation module that carries geometric information inferred from the satellite view as constraints for ground-view generation. Thirdly, we utilize a novel weighted mask to alleviate small misalignment between the satellite-derived depth image and the geo-tagged google panoramic image. Lastly, to the authors' best knowledge, our work is the first cross-view synthesis work that preserves the authentic geometric shape of the ground objects.

2. Related Work

Aerial-Ground Co-location. Aerial-ground co-location is a relevant topic that considers image-level matching cross different views. It is natural that under the cross-view condition, the most reasonable features to be utilized would be semantic information. Castaldo *et al*. [2] proposed to take advantage of available semantic information from GIS maps for matching ground-view images based on the extracted semantic information, and feature descriptors extracted in their common regions (e.g. road intersections) are then matched. Considering that the manually crafted descriptors might not be robust enough, Lin *et al*. [9] proposed to use the deep convolution neural networks to learn feature descriptors from both views, where separate networks respectively for ground and aerial views are trained and simple KNN (k-nearest neighborhood) were applied for potential matches, which was shown to be effective to match local ground-view images over a large aerial image database. Similar ideas were proposed by Hu *et al*. [5] using a learning-based method for localizing panoramic images on larges-scale satellite images, where a two-section Siamese network, namely local feature and global descriptor network were proposed to yield reliable features for matching. Other similar works can be found in [21, 22, 20]. Essentially, these works invariably learn the geometric patterns in the overlapping region, for example, road patterns and junctions. More recently, other approaches such as conditional GANs have been utilized to synthesize views for cross-view matching. For example, in the work of [17], an aerial view was synthesized given a ground-view panorama. This synthesized aerial view was then combined with a ground-view panorama in order to learn feature representations for matching aerial images.

View Transformations. One of the core components in cross-view synthesis and matching is the transformation between different views. Often geometric information is not available and thus learned transformation was presented as an alternative. A few works directly learn such association by taking input of aerial images and outputting both ground-view RGB and semantic images [16, 5], while such methods might only be able to handle scenes with a fixed or similar layout. Instead of embedding the transformation in the feature extraction network, Zhai *et al*. [23] explicitly proposed a module that learned independent transformation between semantic layout between aerial and ground images, thus the transformation between scene labels can be performed in a non-linear fashion. This method works well with landscape scene layout and while is incapable of handling geometric transformation in complex urban scenes, where occlusions and non-planar projective transformation exist.

Cross-view Synthesis. To synthesize views from a completely different scene, although an intuitive idea is to perform image composition through paired image databases, the most recent works are primarily based on generative networks, and more specifically, generative adversarial networks [3] (GANs) are among the most popular. Regmi *et al*. [15] generated aerial or ground view images conditioned

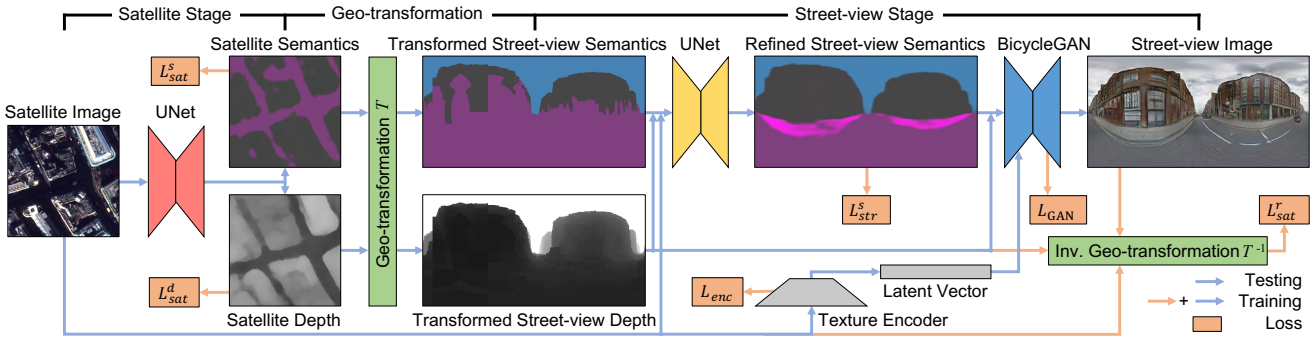


Figure 2: **Overview of our network architecture.** Our network operates in *three* different stages accounting for different image domains. A differentiable geo-transformation stage transforms between the satellite image domain and the street-view image domain. **Satellite Stage:** A U-Net [18] computes for a given satellite image a depth image and corresponding semantic image. **Geo-transformation Stage:** The geo-transformation layer takes the depth and semantic satellite images and transforms them into corresponding depth and semantic panoramic images. **Street-view Stage:** A second U-Net refines the semantic panorama. Finally, a BicycleGAN [28] is used to generate photo-realistic images from the semantic panorama. Rather than using a random seed for the texture generation, we added a separate texture encoder that computes a latent vector from the input satellite image. Symbols in this figure are different types of losses explained in Sec. 3.4.2.

to the other views. This can be performed as image-to-image translation [6] once information from the cross-view such as the scene layout can be estimated [23]. GANs learn a data distribution and when given a perturbed input, they generate data samples following that distribution. Since the ground scene layout predicted from a satellite view through a learning-based transformation can be quite similar, the diversity of generated images can be a challenge.

3. Method

In this section, we introduce our framework for realistic street-view image synthesis from a satellite image which is shown in Fig. 2. Our key idea is to transform the satellite information to the street-view in a geometrically meaningful way for better street-view synthesis. To this end, we use a cascaded architecture with three stages: a satellite stage, a geo-transformation stage, and a street-view stage. The first stage estimates both depth and semantic images from an input satellite image. The second stage transforms the satellite depth and semantic images to street-view via a novel geo-transformation layer. In the last stage, the transformed images are utilized to generate the final street-view images. All three stages are detailed in the following subsections.

3.1. Satellite Stage

This stage follow the assumption that a rough depth image can be obtained from a single image via a convolutional neural network, which has been well investigated in the field of single image depth estimation [8]. We can further exploit the approximately orthogonal projection type of satellite images and that many faces in urban areas are perpendicular to the ground [12]. In addition, semantic labels are also easily obtained [18]. Motivated by previous work [12], we take a U-Net architecture [18] as shown in Fig. 2. In contrast to the network in [12], we utilize a weight-shared decoder to learn both the depth and semantic for the satellite image. In our network, both the encoder and decoder have eight layers, while there are two branches for the last two layers of the decoder, which output the predicted satellite depth image and semantic image, respectively.

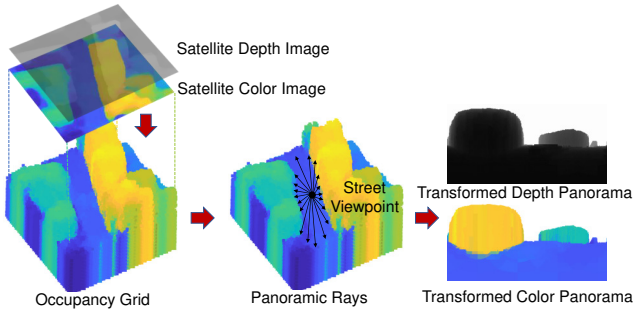


Figure 3: **Illustration of the proposed differential geo-transformation T.** The satellite depth and semantic labels are transformed into a semantic height map, which is then projected into the panoramic image to obtain depth and semantic labels in the street-view domain.

3.2. Geo-transformation Stage

To synthesize a cross-view image, we transform the depth and semantic information from the satellite view to the street-view in a differentiable way such that the whole pipeline can be trained end-to-end. To achieve this goal, we propose the following differentiable geo-transformation workflow as shown in Fig. 3. Given an $n \times n$ square satellite depth image patch $\mathbf{D}$ and a corresponding color image patch $\mathbf{C}$, we perform three steps to obtain a street-view panorama.

Height map generation. First, we transform the satellite depth image into a ground-based height map using orthographic projection.

Occupancy grid generation. Second, this height map is discretized into an $n \times n \times n$ occupancy grid $\mathcal{G}$ by checking for each voxel if it is above or below the corresponding height value. The grid is centered around the street-view location and besides the height value, we also store corresponding RGB values from $\mathbf{C}$ in the voxel grid $\mathcal{G}$ (see Fig. 3).

Panoramic projection. Third, a street view panorama is generated by considering panoramic rays starting from the central voxel of $\mathcal{G}$ and directed to different viewing angles (θ, ϕ) which are transformed into a 3D directional vector:

$$(v_x, v_y, v_z) = (\cos\theta \sin\phi, -\cos\theta \cos\phi, \sin\theta). \quad (1)$$

Then, to generate a $k \times 2k$-sized street-view panorama, we evenly sample $2k$ longitude angles ranging in $\theta \in [0, 2\pi]$ and k latitude angles ranging in $\phi \in [0, \pi]$, which results in $k \times 2k$ panoramic rays shooting into 3D space. We use the panoramic rays to generate a street-view depth panorama. The depth for each pixel in the panorama is determined by the distance between the ray origin and the first encountering voxel (FEV) in the occupancy grid $\mathcal{G}$ along the ray. We sample n 3D points along each ray in equal distances according to the voxel size and then compute the depth by searching the first non-zero voxel value in the occupancy grid $\mathcal{G}$ along the ray.

The street-view color panorama can be generated in the same way. Fig. 3 illustrates the processing pipeline of our geo-transformation procedure. We used $n = 256$ and $k = 256$ in our experiments. Both the 2D to 3D as well as the 3D to 2D transformation procedure is differentiable.

3.3. Street-view Stage

In the street-view stage, we first adopt a U-Net [18] to generate refined semantic labels from the transformed panoramas, and then use a BicycleGAN [28] to translate the semantic labels into an RGB image. As depicted in Fig. 2, the input of the refinement network consists of transformed depth, semantics, and a resized satellite image, which are concatenated together. Then, we concatenate the refined street-view semantics and transformed depth together and feed them into BicycleGAN as the input. The reason why we use BicycleGAN instead of the conventional cGAN [11] is that the translation between semantics and RGB images in our setting is kind of a *multi-modal* image-to-image translation, as two street views which look very different may have similar semantics as long as their structures (e.g. the shape of skyline, the location of sidewalk, etc.) are similar. The generative modeling setting of cGAN cannot address this kind of ambiguity during training. In BicycleGAN, a low-dimensional latent vector is introduced in order to distill the ambiguity. Its generator learns to map the given input, combined with this latent code, to the output. The encoded latent vector is injected by spatial replication and concatenated into every intermediate layer in the encoder. With the latent code, the network is able to produce more diverse results. Nevertheless, the latent vector in the BicycleGAN is originally randomly sampled from a learned distribution during the inference phase. Rather than generating multiple street views our goal is to generate one that is as realistic as possible. Thus, as shown in Fig. 2, we introduce an external encoder which generates such a latent vector from the original satellite image. More details of the sub-networks can be found in the supplementary material.

3.4. Implementation Details

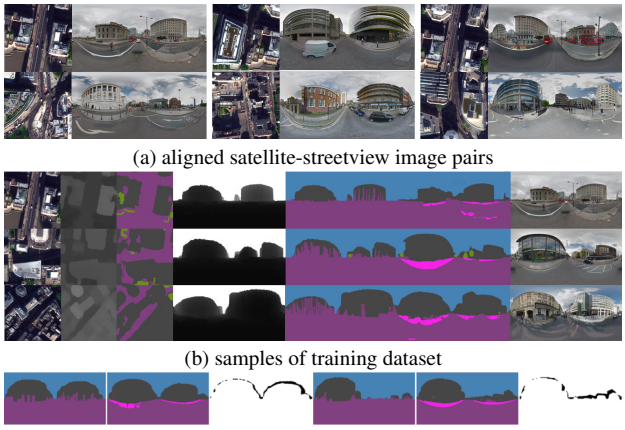
3.4.1 Dataset

For the satellite image, we select a 5km×5km area centered in the city of London as the region of interest. The ground truth depth and semantic satellite images are generated from stereo matching [4, 14, 13] and supervised classification [24] with post corrections, respectively. For the street-view images, we download all available google street-view images in this region via the Google API[1], which results in almost 30K street-view panoramas in total. Each of these panoramas includes location information (longitude, latitude, orientation). However, this GPS information contain certain positional errors, meaning that directly aligning the satellite image using the GPS information of the street-view images typically results in misalignment. In order to mitigate misalignments, we propose a pre-processing strategy to pick out those well-aligned image pairs by calculating their overlap ratios as follows. Firstly, the semantic segmentation result of a real street-view image is obtained by applying SegNet [1]. Subsequently, the sky pixels in this semantic image are compared with those in the street-view semantic image transformed from the corresponding satellite depth image to calculate their overlapping ratio. The image pairs with an overlapping ratio greater than 90% are kept as a well-aligned training dataset. In this way, we obtained approximately 2K well-aligned satellite-street-view image pairs. Fig. 4 shows examples of our training dataset.

3.4.2 Loss Function

The overall loss function of our full pipeline is defined as $L = L_{sat} + L_{str}$, representing the losses of satellite stage and street-view stages respectively. The satellite loss consists of two terms, $L_{sat} = L^d_{sat} + L^s_{sat}$, representing the L_1 losses for the satellite depth image and semantics respectively. The street-view loss is composed of four terms $L_{str} = L^s_{str} + L_{\text{GAN}} + L_{enc} + L^r_{sat}$, representing a weighted L_1 loss for the street-view semantics, the BicycleGAN loss (consisting of the L_1 losses for the generator and the latent vector encoder, and log-losses for the 2 discriminators), an L_1 loss for the external encoder and an L_1 loss for

[1] https://developers.google.com/maps/documentation/streetview/intro



(a) aligned satellite-streetview image pairs

(b) samples of training dataset

(c) samples of misaligned training samples

Figure 4: **Examples of our training datasets.** For (a), from left to right are the satellite image and the corresponding street-view image. For (b), from left to right are the satellite image, satellite depth, satellite semantic, transformed street-view depth, transformed street-view semantic, true street-view semantic, and true street-view RGB, respectively. For (c), from left to right are the transformed street-view semantic, true street-view semantic, and the misaligned mask.

the predicted satellite RGB produced by the inverted geo-transformation, respectively.

The reason why we adopt a weighted L_1 loss for the street-view semantics is to deal with the misalignment between the satellite image and street-view image (as introduced in Sec. 3.4.1). The weighted L_1 loss is defined as $L_1^W = L_1(W * ||I - I_{\text{GT}}||)$, where W is a weight matrix which controls the weight for each pixel and the sign $*$ represents the element-wise multiplication. The weight matrix is designed to give less weight to misaligned pixels. As Fig. 4 (c) shows, misaligned pixels usually occur along the boundaries between sky and buildings, where the sky pixels may be incorrectly labeled as building and vice versa. We reduce the loss for these mislabeled pixels to 10% of the loss of the remaining pixels.

The loss of the inverted geo-transformation L_{sat}^r is designed to make the road pixels in the predicted street-view image as similar as possible to the road pixels in the input satellite image. Given the transformed street-view depth panorama I_{prj}^d, the transformed street-view RGB panorama I_{pred}^r, and the corresponding satellite RGB image I_{sat}^r, the loss L_{sat}^r is computed in the following four steps. Firstly, the panoramic ray for each pixel on I_{prj}^d is calculated as in Eq. (1). Then, a 3D point (x, y, z) for each pixel (i, j) in I_{prj}^d can be calculated as:

$$\begin{aligned} x &= v_x(i,j) \cdot I_{prj}^d(i,j) + x_c, \\ y &= v_y(i,j) \cdot I_{prj}^d(i,j) + y_c, \\ z &= v_z(i,j) \cdot I_{prj}^d(i,j), \end{aligned} \tag{2}$$

where $\big(v_x(i,j), v_y(i,j), v_z(i,j)\big)$ is the normalized

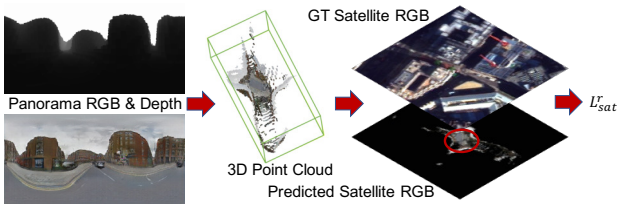


Figure 5: **Visualization of the inverse geo-transformation T^{-1}.** We consider only the pixels within a 5m radius around the location (marked by the red circle) for the L_{sat}^r loss.

panoramic ray for pixel (i, j), $(x_c, y_c) = (\frac{wg}{2}, \frac{hg}{2})$ is the (x, y) coordinate of the central pixel in the satellite image image, which is (64,64) for constant since the size (w, h) for the satellite image is (256,256) and the ground sampling distance g is 0.5. After that, the RGB values of the 3D points are picked from the predicted street-view image and saved into an RGB image with the same size as I_{sat}^r, which forms the inverted transformed satellite image. Finally, considering that only the road pixels can be observed in both satellite and street-view images, the L_1 loss is calculated just for pixels within a 5m (10 pixels) range to the center of the inverted transformed satellite image with I_{sat}^r. Fig. 5 gives a brief demonstration on the workflow of the inverted geo-transformation loss.

3.4.3 Training Scheme

Due to the cascaded architecture and the location misalignment problem, we first pre-train each stage of our pipeline independently, and then fine-tune our network in an end-to-end manner. We train our model on a computer with Intel CPU i7, 16GB RAM and an Nvidia GeForce GTX1080 GPU. The full pipeline is implemented with PyTorch. For all network trainings, we used Adam [7] as the optimizer with an initial learning rate of 2×10^{-3}, $\beta_1 = 0.5$. The learning rate is decreased by half every 100 epochs.

For the network of the satellite stage, we crop a 256$\times$ 256 patch for each of the 30K panoramas by converting the longitude and latitude of the panorama on to the satellite images and choose 10K among them to train the satellite image to depth and semantic label network. Some training samples for this stage can be found in the first three columns of Fig. 4 (b). This network was trained for 200 epochs.

For the transformed semantics refinement network in the street-view stage, we utilize the 2K aligned satellite-streetview image pairs for training. The ground truth semantic label of the street-view image is obtained by applying SegNet [1] directly on our street-view images, which results in a semantic image with dozens of labels. We further merged some of these labels to form a clean semantic image with only 4 classes: sky, building, sidewalk, and ground. Some training samples for this network can be found in the 4th to 6th columns in Fig. 4 (b). We trained this network

for 50 epochs because the mapping from the input to the output is relatively simple, also more epochs can lead to over-fitting in practice.

For the final street-view image generation network, we use the same 2K image pairs. Also, the default training settings of BicycleGAN are employed except for the dimension of the latent vector, which we set to 32, and the basic feature dimension was set to 96. The external texture encoder has the same architecture as the encoder in BicycleGAN. We first train the network on randomly cropped training pairs for 400 epochs and then train on full image pairs for 50 epochs.

4. Experiments

4.1. Baselines and Evaluation Metrics

Regmi *et al.* [15] proposes two cGAN-based architectures to generate the street-view semantic and RGB image given an aerial image patch as input. The "fork architecture" which uses a weight-shared decoder for the simultaneous generation of both the semantic image and the RGB image has been shown to be better than the other "sequence architecture" which uses a sequential network generating semantic image first and then the RGB image. We utilized the original code and compared it with the "fork architecture".

Pix2Pix [27, 6] is a well known cGAN-based network which can also be utilized to synthesize street-view images from the satellite images. Therefore, we also compared to this method using the original source code.

For quantitative assessment we utilize various **evaluation metrics** ranging from low-level to high-level. For the low-level metrics, we follow [15] and use PSNR, SSIM, and Sharpness Difference metrics, which evaluate the per-pixel difference between the predicted image and the ground truth image. However, such pixel-wise metrics might not properly assess the visual quality of the images. Therefore, we use the perceptual similarity [25] to compare the images on a higher semantic level. P_{Alex} and P_{Squeeze} denote the evaluation results based on the backbone of AlexNet and SqueezeNet, respectively. We directly employ their code and the provided pre-trained model. For the semantic-level metrics we use the pixel-level accuracy and mIoU from [10], which is calculated by comparing the semantic labels of the predicted street-view image and the ground truth image generated using SegNet [26]. For the geometric-level metric we utilize the boundary F-score [19] which is designed to calculate the overlap between the object boundaries in the predicted and the ground truth images. We also compute the median error e_{depth} of the generated panorama depth by taking the depth computed from satellite multi-view stereo reconstruction as the ground truth.

In the following we present a state-of-the-art comparison and an ablation study in Sec. 4.2 and 4.3. For more experimental results, please refer to our supplementary material.

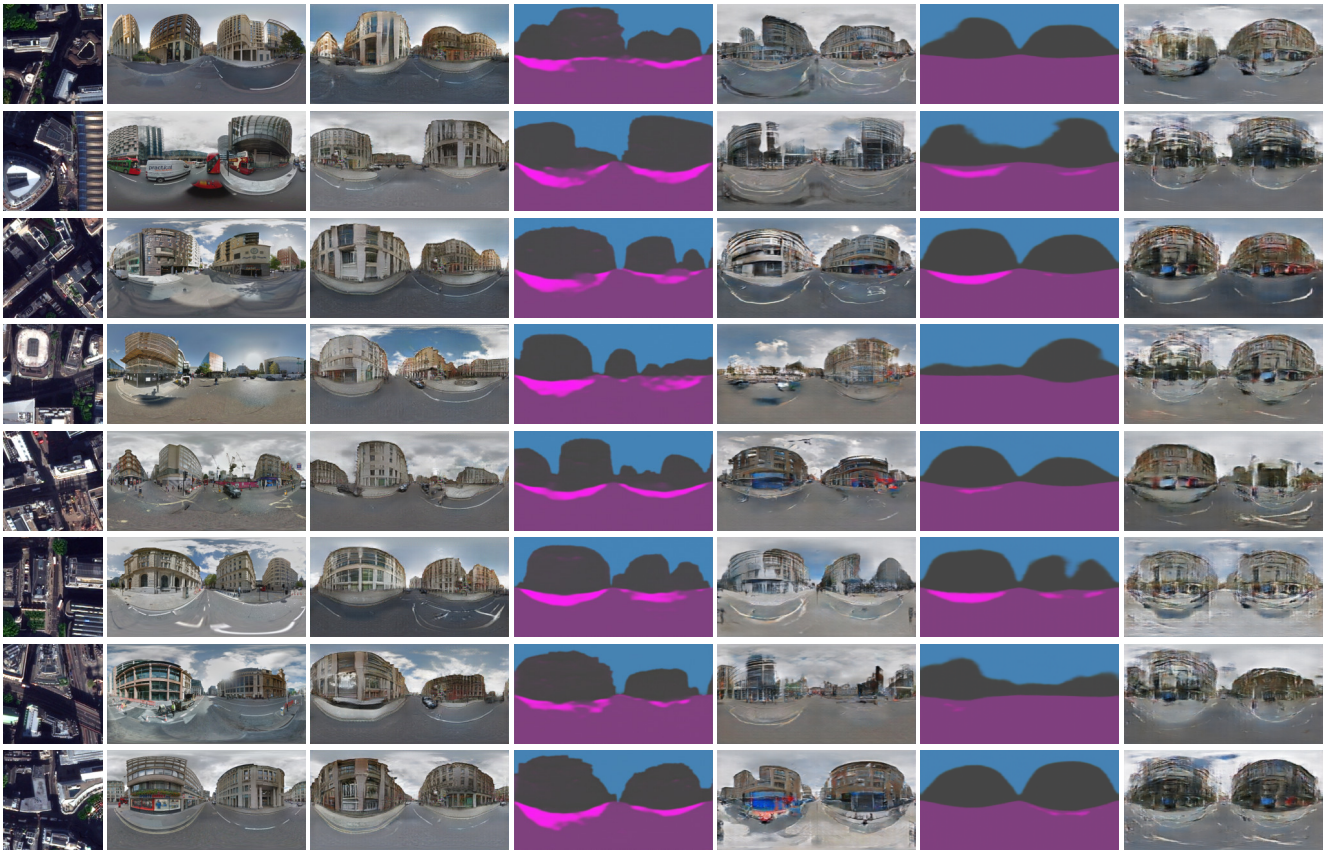
4.2. Comparison to State of the Art

Tab. 1 provides quantitative evaluation results for Pix2Pix [6], Regmi *et al.* [15], and our method in a testing dataset with 100 samples. Due to the fact that we use more problem-specific knowledge, our method outperforms all competing approaches on all measures. As other approaches cannot generate depth images, we only evaluate the median depth error of our method, which is 2.62m. We use the same quantitative evaluation measures as in [15] and we can see in Tab. 1 that there is little difference between the PSNR, SSIM, and Sharpness Difference (larger is better) of the three methods, which is reasonable since we have analyzed that the low-level metrics can hardly be utilized to judge whether an image is realistic or not.

Fig. 6 shows the qualitative results of these three methods. From the figure, we can observe that the quality of the generated semantic and RGB images of our method is better than the other two methods. Firstly, for the street-view semantic image, it is obvious that the semantic image of the work of Regmi *et al.* is a relatively coarse prediction of the street-view layout, which may contain significant artifacts in some cases (e.g. row 2,3 and 5 in Fig. 6). While for our proposed method, the street-view layout is very close to the ground truth because our geo-transformation layer can transform the true geometric information from the satellite to the street-view. Also, the estimated position of the sidewalk in the result of Regmi *et al.*'s method appears to be randomly generated as it does not show many patterns on which building of the scene the sidewalk might appear, and the sidewalks in our cases are fairly consistent and can be detected as long as there are buildings. Secondly, for the quality of the generated RGB image, our method also demonstrates its advantages over the other two methods: Regmi *et al.* and Pix2Pix, and this should be largely credited to the high quality of the generated semantic images in our pipeline. The result of Regmi *et al.*'s work is slightly better than that of Pix2Pix in terms of that Regmi *et al.*'s work can generate more complete images. However, the images generated by both of the two methods are blurred in terms of their texture details and only part of the geometric configuration can reflect the actual scene the satellite image captures. Fig. 8 further compares the detailed geometry information of images generated by our method and the state-of-the-art approaches. We can find that our method can better recover the building skyline shape. We also noticed that the fine-detailed objects like trees and buses cannot be reconstructed, which is mainly because moving objects (e.g. buses) and fine-detailed static objects (e.g. trees) cannot be reconstructed well in the satellite depth and are also inconsistent in the cross-view images.

Table 1: **Quantitative evaluation of image/semantic quality.** Our method consistently outperforms competing approaches.

Method	PSNR (↑)	SSIM (↑)	Sharp Diff (↑)	P_{Alex} (↓)	$P_{Squeeze}$ (↓)	mIoU (↑)	Acc. (↑)	F_{sem} (↑)	e_{depth} (↓)
Pix2Pix [6]	19.765	0.410	20.514	0.6062	0.4778	0.371	0.434	0.445	N/A
Regmi *et al.* [15]	19.839	0.419	20.559	0.5867	0.4430	0.484	0.649	0.486	N/A
Ours	**19.943**	**0.440**	**20.864**	**0.5816**	**0.4339**	**0.548**	**0.729**	**0.515**	2.62



Sat. RGB | Street RGB | Our RGB | Our Semantics | [15] RGB | [15] Semantics | [6] RGB

Figure 6: **Qualitative comparison.** We present a variety of test results of our method, in comparison to Regmi *et al.* [15], and Pix2Pix [6]. Our method generates significantly more accurate semantic maps, especially with respect to the skyline, but also our RGB output looks more realistic and contains fewer artifacts.

4.3. Ablation Study

We further investigate the influence of multiple key components on the performance of our approach and the detailed quantitative result can be seen in Tab. 2. In the following, we study the impact of three network components.

Importance of input with depth. In theory, depth can provide "scale" information in the generation of local texture. Comparing street-view images generated by our method (Fig. 7 (d)) and our method w/o depth (Fig. 7 (e)), we can find that the textures of the objects close to the camera center in the real scene on those images are well generated. While for those objects far away from the camera center, the method w/o depth can only generate rough and blurred textures. That explains why the semantic mIoU and accuracy do not drop too much while the perception distance increases significantly since the semantic label of the blurred texture will still be correct but the perception distance of it will increase. As a result, we can conclude that without depth one may not generate detailed texture in the distant areas but still able to get quite good semantics.

Importance of weighted L_1 loss. As mentioned in Sec. 3.4.2, the weighted L_1 loss for L_{sat}^s is designed to reduce the influence of the misalignment problem and improve the quality of the generated street-view semantic images. To evaluate the importance of the weighted L_1 loss, we trained our pipeline w/o the weighted L_1 loss and output the generated street-view semantic image (Fig. 7 (b)) and RGB image (Fig. 7 (f)) for comparison. As can be observed, there are some misclassified pixels between the building class and the road class on the semantic image w/o weighted L_1 loss, also the boundary between the building roof the sky is mixed in the second row. The misclassifica-

Table 2: **Quantitative Ablation Study.** We demonstrate the impact of various network components quantitatively.

Method	PSNR (↑)	SSIM (↑)	Sharp Diff (↑)	P_{Alex} (↓)	$P_{Squeeze}$ (↓)	mIoU (↑)	Acc. (↑)	F_{sem} (↑)
Ours	19.943	**0.440**	**20.864**	**0.5816**	**0.4339**	**0.548**	**0.729**	0.515
w/o depth	19.991	0.419	20.783	0.6523	0.4539	0.537	0.728	**0.534**
w/o weighted L_1 loss	**20.170**	0.433	20.711	0.5818	0.4364	0.535	0.727	0.505
w/o geo-transformation layer	20.002	0.401	20.459	0.6518	0.4548	0.509	0.711	0.504

(a) Predicted semantics (b) w/o weighted L_1 loss (c) Ground truth (d) Ours (e) w/o depth (f) w/o weighted L_1 loss (g) w/o geo-transf.

Figure 7: **Qualitative Ablation Study.** In correspondence to the quantitative ablation study in Tab. 2 we show example result images for each configuration. Omitting one of the components typically results in worse results.

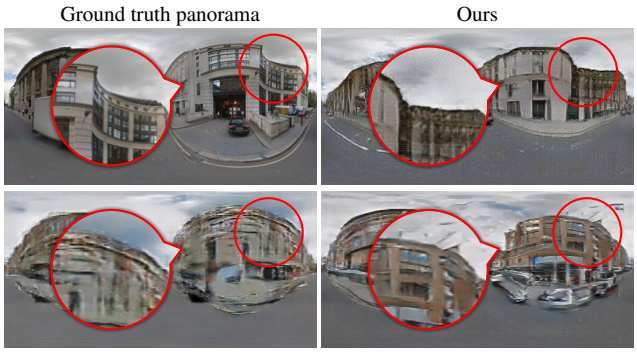


Figure 8: **Comparison of results on skylines.** Due to the explicit use of geometry information, our method estimates more accurate skyline profiles. Although the competing methods show results for the same location, the appearance is different mostly due to the incorrect skyline estimation.

tion on the semantic image further impairs the quality of the generated RGB image as shown in the second row. However, the misclassification on the semantic image caused by the misalignment problem can somehow be relieved in the following network for street-view RGB image generation due to the power of CNN, which the CNN network can tolerate some misclassification. This is a reason why the quantitative scores in Tab. 2 are not too different.

Importance of the geo-transformation layer. We remove the geo-transformation layer in our pipeline and feed the outputs of the satellite stage directly into the street-view stage to see the influence of the geo-transformation layer. From Tab. 2, we can see that the semantic mIoU and accuracy drop significantly, while the perceptual score P_{Alex} increases from 0.5816 to 0.6518. This means that both the semantic and the perceptual quality of the RGB image generated w/o the geo-transformation layer decrease largely. This observation is further supported by Fig. 7 (g), where the buildings generated in the first and third rows are largely distorted from the ground truth images as shown in Fig. 7 (c). Therefore, the generation of street-view semantics directly from the predicted satellite depth and semantic images is more likely to yield geometrically incorrect results than applying the proposed geo-transformation approach.

5. Conclusion

We presented a novel approach for satellite-to-ground cross-view synthesis. In particular, we proposed an end-to-end trainable pipeline that takes a single satellite image and generates a geometrically consistent panoramic RGB image. We thus proposed a neural network with a differentiable geo-transformation layer that links a semantically labeled satellite depth image with a corresponding semantic panoramic street-view depth image which is finally used for photo-realistic street-view images generation. The geometric consistency across images significantly improves the accuracy of the skyline in the panoramic ground view which is especially important for urban areas. Our experiments demonstrate that our method outperforms existing approaches and is able to synthesize more realistic street-view panoramic images and in larger variability.

Acknowledgments. Zuoyue Li received funding by a Swiss Data Science Center Fellowship. Zhaopeng Cui was supported by DSO National Laboratories within the project AutoVision. The Multi-view Satellite Images are acquired from DigitalGlobe. Further, this research was partially supported by the Office of Naval Research (Award No. N000141712928) and by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior/ Interior Business Center (DOI/IBC) contract number D17PC00280. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DOI/IBC, or the U.S. Government.

References

[1] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(12):2481–2495, 2017. 4, 5

[2] Francesco Castaldo, Amir Zamir, Roland Angst, Francesco Palmieri, and Silvio Savarese. Semantic cross-view matching. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 9–17, 2015. 2

[3] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2672–2680, 2014. 2

[4] Heiko Hirschmuller. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):328–341, 2008. 4

[5] Sixing Hu, Mengdan Feng, Rang MH Nguyen, and Gim Hee Lee. Cvm-net: Cross-view matching network for image-based ground-to-aerial geo-localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7258–7267, 2018. 2

[6] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1125–1134, 2017. 3, 6, 7, 8

[7] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations*, 2015. 5

[8] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2041–2050, 2018. 3

[9] Tsung-Yi Lin, Yin Cui, Serge Belongie, and James Hays. Learning deep representations for ground-to-aerial geolocalization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5007–5015, 2015. 2

[10] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015. 6

[11] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets, 2014. cite arxiv:1411.1784. 4

[12] Lichao Mou and Xiao Xiang Zhu. Im2height: Height estimation from single monocular imagery via fully residual convolutional-deconvolutional network. *arXiv preprint arXiv:1802.10249*, 2018. 3

[13] Rongjun Qin. Rpc stereo processor (rsp)–a software package for digital surface model and orthophoto generation from satellite stereo imagery. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 3:77, 2016. 4

[14] Rongjun Qin. Automated 3d recovery from very high resolution multi-view satellite images. In *ASPRS (IGTF) annual Conference*, page 10, 2017. 4

[15] Krishna Regmi and Ali Borji. Cross-view image synthesis using conditional gans. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3501–3510, 2018. 2, 6, 7, 8

[16] Krishna Regmi and Ali Borji. Cross-view image synthesis using geometry-guided conditional gans. *arXiv preprint arXiv:1808.05469*, 2018. 2

[17] Krishna Regmi and Mubarak Shah. Bridging the domain gap for ground-to-aerial image matching. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019. 2

[18] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-assisted Intervention*, pages 234–241. Springer, 2015. 3, 4

[19] Towaki Takikawa, David Acuna, Varun Jampani, and Sanja Fidler. Gated-scnn: Gated shape cnns for semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5229–5238, 2019. 6

[20] Yicong Tian, Chen Chen, and Mubarak Shah. Cross-view image matching for geo-localization in urban environments. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3608–3616, 2017. 2

[21] Nam N Vo and James Hays. Localizing and orienting street views using overhead imagery. In *European Conference on Computer Vision*, pages 494–509. Springer, 2016. 2

[22] Scott Workman, Richard Souvenir, and Nathan Jacobs. Wide-area image geolocalization with aerial reference imagery. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3961–3969, 2015. 2

[23] Menghua Zhai, Zachary Bessinger, Scott Workman, and Nathan Jacobs. Predicting ground-level scene layout from aerial imagery. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 867–875, 2017. 2, 3

[24] Qian Zhang, Rongjun Qin, Xin Huang, Yong Fang, and Liang Liu. Classification of ultra-high resolution orthophotos combined with dsm using a dual morphological top hat profile. *Remote Sensing*, 7(12):16422–16440, 2015. 4

[25] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018. 6

[26] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 6

[27] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2223–2232, 2017. 6

[28] Jun-Yan Zhu, Richard Zhang, Deepak Pathak, Trevor Darrell, Alexei A Efros, Oliver Wang, and Eli Shechtman. Toward multimodal image-to-image translation. In *Advances in Neural Information Processing Systems*, 2017. 3, 4