IMPLICITY: CITY MODELING FROM SATELLITE IMAGES WITH DEEP IMPLICIT OCCUPANCY FIELDS

Corinne Stucker*, Bingxin Ke, Yuanwen Yue, Shengyu Huang, Iro Armeni, Konrad Schindler
ETH Zurich, Switzerland – (stuckerc, bingke, yuayue, shenhuan, iarmeni, schindler)@ethz.ch

KEY WORDS: 3D Reconstruction, Digital Surface Model (DSM), Deep Implicit Fields, Scene Representation, Satellite Imagery.

ABSTRACT:

High-resolution optical satellite sensors, combined with dense stereo algorithms, have made it possible to reconstruct 3D city models from space. However, these models are, in practice, rather noisy and tend to miss small geometric features that are clearly visible in the images. We argue that one reason for the limited quality may be a too early, heuristic reduction of the triangulated 3D point cloud to an explicit height field or surface mesh. To make full use of the point cloud and the underlying images, we introduce IMPLICITY, a neural representation of the 3D scene as an implicit, continuous occupancy field, driven by learned embeddings of the point cloud and a stereo pair of ortho-photos. We show that this representation enables the extraction of high-quality DSMs: with image resolution 0.5 m, IMPLICITY reaches a median height error of $\approx$ 0.7 m and outperforms competing methods, especially w.r.t. building reconstruction, featuring intricate roof details, smooth surfaces, and straight, regular outlines.

1. INTRODUCTION

Modern very high-resolution (VHR) satellite sensors have made it possible to reconstruct sub-meter resolution 3D surface models from space. They are able to collect optical images with ground sampling distances $\leq$0.5 m from multiple viewpoints almost anywhere on Earth. Several software packages have been developed to derive 3D models from such satellite images (Krauß et al., 2013, De Franchis et al., 2014, Qin, 2016, Rupnik et al., 2017, Beyer et al., 2018, Cournet et al., 2020, Youssefi et al., 2020). Typically, they adopt stereo matching algorithms originally developed for terrestrial or airborne photogrammetry. The principle of such algorithms is to find a dense set of image correspondences that have high photo-consistency and at the same time form a (piece-wise) smooth surface. After matching all suitable image pairs, the correspondences are triangulated to 3D points and fused into a single point cloud, which is commonly rasterized into a 2.5-dimensional height field (a.k.a. digital surface model, DSM) for further use.

Due to limited image resolution, sub-optimal stereo geometry, and radiometric differences caused by variable lighting and atmospheric effects, DSMs derived from satellite observations tend to be noisy (see Figure 1). Moreover, high-frequency details that would, in principle, be visible in the images are barely reconstructed. Those DSMs are thus often regarded as intermediate products and processed further, with a refinement step that aims to suppress noise and to impose a-priori assumptions about the surface, like straight building edges and vertical walls. Early attempts used low-level filtering and hand-coded rules. More recent works rely on neural networks to learn the mapping from a coarse DSM to a refined one from data (Bittner et al., 2019b, Bittner et al., 2019a, Bittner et al., 2020, Wang et al., 2021, Stucker and Schindler, 2022).

A fundamental property shared by different DSM reconstruction and refinement methods is an explicit representation of the surface, either as a mesh with a given number of vertices (respectively, faces) or as a regular 2D grid of height values. Such explicit parametrizations are convenient, but they do not preserve all information contained in the original point cloud and restrict the ability to resolve small structures. Recently, implicit neural functions have emerged as a powerful and effective representation of 3D geometry (Park et al., 2019, Chen and Zhang, 2019, Mescheder et al., 2019, Peng et al., 2020). Instead of discretizing the 3D scene into a set of explicit surface elements, they implicitly model its geometry as a continuous field of occupancies or signed distance values, encoded in the weights of a neural network. The network can be evaluated at any 3D coordinate and, therefore, conceptually, allows for infinite resolution—in practice, its effective resolution is bounded by the representation power of the finite number of neurons, as well as by the resolution of the training data.

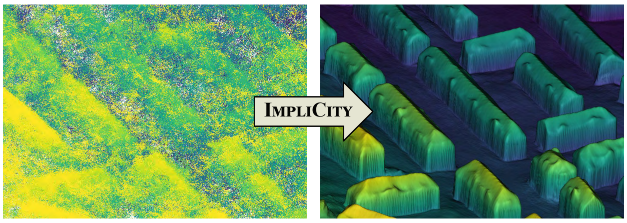

Figure 1. IMPLICITY is a deep, implicit representation of surface geometry. It is derived from a photogrammetric 3D point cloud and associated images. Note the geometric details on the roofs despite the substantial noise level of the input point cloud.

So far, implicit representations have been explored to model the 3D geometry of local shapes (Genova et al., 2019, Genova et al., 2020), single objects (Park et al., 2019, Atzmon and Lipman, 2020), indoor scenes (Jiang et al., 2020, Peng et al., 2020, Sitzmann et al., 2020, Chabra et al., 2020), and single buildings (Chen et al., 2021). In this work, we go one step further and investigate their potential to accurately reconstruct 3D urban scenes, on the order of several km^2, from satellite data. To that end, we introduce IMPLICITY, a coordinate-based, implicit neural 3D scene representation based on a point cloud derived from satellite photogrammetry. Since such point clouds are comparatively sparse and lack high-frequency detail, we additionally use an image stereo pair to guide the occupancy prediction. IMPLICITY reconstructs city models with fine-grained shape details, smooth and well-aligned surfaces, and crisp edges. It thereby reduces the mean absolute error by >60% compared to a conventional stereo DSM.

2. RELATED WORK

Deep Implicit Functions. Deep implicit functions for surface reconstruction have been proposed concurrently by (Mescheder

* Corresponding author

et al., 2019, Park et al., 2019, Chen and Zhang, 2019). These seminal works represent a 3D shape as an implicit, continuous field f, which is parametrized as a neural *decoder* network, and constrained by a global latent *code* (a "feature vector" of the scene) extracted with neural *encoder* network. The field f can be queried with a 3D location $\mathbf{x} \in \mathbb{R}^3$ and returns either the *occupancy* of $\mathbf{x}$ (i.e., its probability of lying below the surface) or its signed distance to the surface. To extract an explicit surface model, one reconstructs the iso-surface $f = 0.5$ of the occupancy, respectively $f = 0$ for the signed distance, for instance with marching cubes (Lorensen and Cline, 1987).

As the scene information is stored as a global latent code, the method described so far does not generalize to unseen objects, fails to capture local surface details, and scales poorly with scene size. Therefore, more recent works (Chabra et al., 2020, Jiang et al., 2020, Genova et al., 2020, Peng et al., 2020) decompose the scene into parts that are constrained by local codes. Moreover, (Peng et al., 2020) introduce a fully convolutional encoder. In this way, the implicit representation inherits the translation equivariance of convolutions; which, in turn, enables large-scale reconstructions. (Saito et al., 2019) introduce local latent codes that are pixel-wise aligned with the image used for supervision, so as to obtain crisp surface edges aligned with the image gradients. The work perhaps most similar in spirit to ours is (Yang et al., 2021), where an implicit neural model is used to reconstruct humans from LiDAR scans, guided by a (single) image to retrieve details such as the wrinkles of clothes.

Deep Implicit Functions for Satellite Images. To the best of our knowledge, (Derksen and Izzo, 2021, Xiangli et al., 2021) are so far the only works that have explored deep implicit representations in the context of satellite data. Both are based on the Neural Radiance Field (NeRF) method of (Mildenhall et al., 2020) that models an observed 3D scene as a continuous, volumetric field of viewpoint-dependent radiance values. The NeRF approach is designed primarily for novel-view synthesis, not geometrically detailed reconstruction. It does, of course, implicitly capture 3D geometry, but with an accuracy just enough to render it from new viewpoints and obtain radiometrically convincing images. E.g., the average reconstruction errors reported in (Derksen and Izzo, 2021) are similar to those of conventional satellite photogrammetry. Also, the NeRF encoder must capture viewpoint-dependent appearance changes, and therefore extract implicit lighting and material information. As a consequence, it cannot generalize beyond the training region.

3. METHOD

Our approach starts from a set of satellite images $\mathcal{I}$ with overlapping fields of view and known camera poses. We follow best practices for satellite photogrammetry and first perform conventional, dense image matching for all suitable image pairs, followed by triangulation. See Section 4.1.

Problem Formulation. Given a set of triangulated 3D points $\mathcal{P} = \left\{\mathbf{p}_i \in \mathbb{R}^3\right\}_{i=1}^{N}$, collected from all stereo pairs, our goal is to build a detailed and geometrically accurate 3D reconstruction of the observed scene. This is where our approach deviates from standard practice: we do not convert the raw point cloud $\mathcal{P}$ into a raster DSM for subsequent 2.5D processing. Instead, we reason in 3D space and represent the scene geometry as a continuous occupancy field. The field is represented by a function f_θ that, for a given 3D coordinate $\mathbf{x} \in \mathbb{R}^3$, returns the probability $\hat{o}$ that the location is occupied. I.e., f_θ should be 0 wherever there is free space, and 1 on and underneath the surface:

$$f_\theta\big(\mathbf{x}, \psi(\mathcal{P}, \mathbf{x}), \xi(\mathcal{I}, \mathbf{x})\big) \rightarrow \hat{o} \in [0, 1] \,, \qquad (1)$$

where $\psi(\mathcal{P}, \mathbf{x}) \in \mathbb{R}^d$ and $\xi(\mathcal{I}, \mathbf{x}) \in \mathbb{R}^d$ are location-dependent latent codes that modulate the occupancy probability. In our case, $\psi(\mathcal{P}, \mathbf{x})$ describes the local structure of the point cloud $\mathcal{P}$, whereas $\xi(\mathcal{I}, \mathbf{x})$ encodes the local image texture at the 2D projections of $\mathbf{x}$ in two input views. Inspired by recent trends in 3D surface reconstruction, we parametrize f_θ as well as the feature extractors $\psi(\mathcal{P}, \mathbf{x})$ and $\xi(\mathcal{I}, \mathbf{x})$ as neural networks. To extract an explicit surface from this implicit representation, one must sample a sufficiently dense set of 3D locations $\mathbf{x}$, evaluate the function f_θ at all of them, and extract the iso-surface $f_\theta = 0.5$.

Overview. Figure 2 depicts an overview of our approach. At its core is IMPLICITY, a coordinate-based neural representation of 3D scene geometry, guided by satellite images. The inputs to IMPLICITY are a raw, irregular, and unoriented point cloud $\mathcal{P}$, as obtained from satellite-based stereo reconstruction, and two ortho-rectified (panchromatic) satellite images. We first map every point in $\mathcal{P}$ to a feature vector that encodes its geometric context, then aggregate those feature vectors into a shape embedding ψ. The embedding ψ is aligned with the geographic coordinates, i.e., its z-axis is the vertical and its (x, y)-axes are the East and North directions in the local UTM zone. Similarly, we map the ortho-images to an image embedding ξ with a fully convolutional 2D encoder, so as to encourage consistency between the reprojected 3D scene geometry and the image content. Note that, since the ortho-images are rectified to UTM coordinates, the shape embedding ψ and the image embedding ξ are, by construction, aligned and share the same (x, y)-axes. With these embeddings, we can, at any location $\mathbf{x}$, read out the two d-dimensional codes $\psi(\mathcal{P}, \mathbf{x}), \xi(\mathcal{I}, \mathbf{x})$ and pass them, together with the coordinates, to a decoder function $f_\theta : \mathbb{R}^3 \times \mathbb{R}^d \times \mathbb{R}^d \rightarrow [0, 1]$ that infers the occupancy state at $\mathbf{x}$. In our case, the decoder is a multi-layer perceptron with internal skip connections.

In the following, we introduce our network architecture (Section 3.1) and its variants (Section 3.2) in more detail before proceeding to the training procedure (Section 3.3) and the sampling strategy employed to define the training signal (Section 3.4). Finally, we describe how we convert the implicit occupancy volume into an explicit raster DSM (Section 3.5).

3.1 Network Architecture

The architecture of IMPLICITY builds upon recent advances in learned, implicit 3D modeling. We adopt the convolutional single-plane encoder proposed by (Peng et al., 2020) to process the input point cloud and the pixel-aligned encoder of (Saito et al., 2019) to process the ortho-images. As a decoder, we apply the same fully-connected network as in (Peng et al., 2020).

Shape Embedding. To represent the local 3D point distribution that forms the basis for surface reconstruction, we compute a feature encoding from the point cloud $\mathcal{P}$. We follow (Peng et al., 2020) and first apply a point-wise encoder based on PointNet (Qi et al., 2017), with one fully connected layer followed by five fully-connected ResNet blocks (He et al., 2016). Each ResNet block includes a local pooling operation to locally aggregate 3D context information. The extracted d-dimensional per-point features are then orthographically projected onto a horizontal plane and discretized into a regular 2D grid of $H \times W$ grid cells, where features that project into the same cell are averaged. In our implementation, we use a grid spacing of 0.5 m in

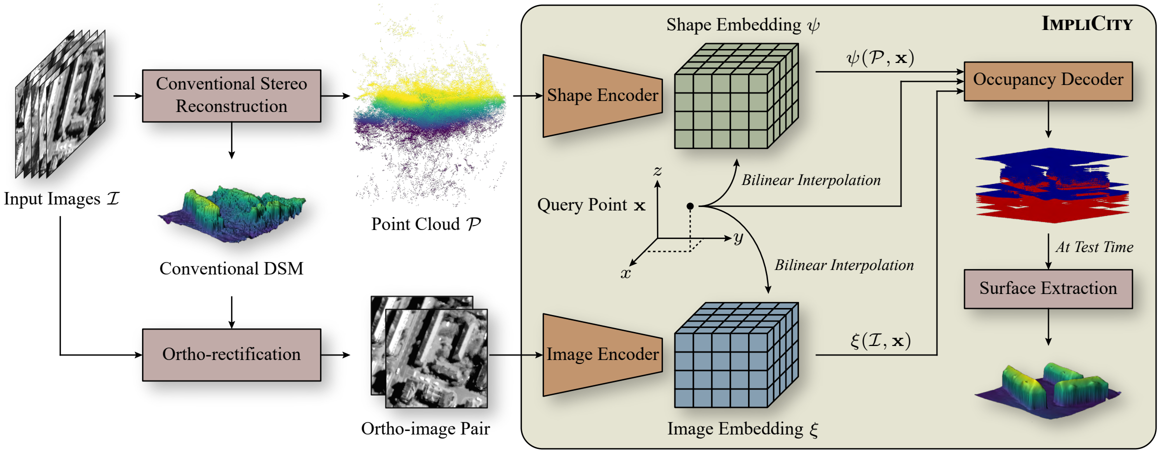

Figure 2. Method overview. Satellite images are processed into a 3D point cloud and a coarse DSM as a basis for ortho-rectification (left side). IMPLICITY takes the point cloud and ortho-photos as input and transforms them into a shape embedding ψ and an image embedding ξ, which can be decoded into occupancy values in continuous 3D space to recover a high-accuracy DSM.

world coordinates. Following (Peng et al., 2020), the resulting feature "image", with size $H \times W \times d$, is processed further with a 2D U-Net (Ronneberger et al., 2015), equipped with symmetric skip connections to preserve high-frequency information. To capture long-range context, the depth of the U-Net is set such that its receptive field spans the entire feature image.

Satellite Image Embedding. Point clouds derived from satellite images are comparatively sparse and fairly noisy (cf. Figure 1). As a consequence, they do not preserve high-frequency details (like sharp roof edges or small dormers) that are, in principle, visible in the images. To recover fine-grained geometric details, we thus build a second latent embedding ξ from a panchromatic stereo pair. That image embedding is then used as additional input to the decoder to guide the occupancy prediction. The two images of the stereo pair are aligned by ortho-rectifying both of them with the same, preliminary surface model (cf. Section 4.1) and stacked into a two-channel image. To generate ξ, we process that image with an encoder similar to the stacked hourglass architecture (Newell et al., 2016) used in PIFu (Saito et al., 2019). To adapt it for our purposes, we modify the first layer to accept our two-channel input and change the hidden feature dimension d to match that of the shape embedding ψ. Note that ortho-rectifying the images *(i)* makes it possible to work with a single image embedding despite the two different viewpoints, and *(ii)* ensures that the embeddings ψ and ξ are correctly aligned.

Occupancy Decoder. The task of the decoder is to estimate the occupancy probability at any location in scene space. Given a point $\mathbf{x} \in \mathbb{R}^3$, we project it onto the horizontal (x, y) coordinate plane and retrieve its shape code $\psi(\mathcal{P}, \mathbf{x})$ and image code $\xi(\mathcal{I}, \mathbf{x})$ from the two embeddings with bilinear interpolation. The occupancy at $\mathbf{x}$, as a function of its coordinates $\mathbf{x}$, shape code $\psi(\mathcal{P}, \mathbf{x})$, and image code $\xi(\mathcal{I}, \mathbf{x})$, is then predicted with a network consisting of five consecutive, fully-connected ResNet blocks. In our implementation, each ResNet block has d neurons, and the sum $\psi(\mathcal{P}, \mathbf{x}) + \xi(\mathcal{I}, \mathbf{x})$ of the two codes is added as side input to every block, as in (Peng et al., 2020).

3.2 Network Variants

In our method, the stereo images are simply stacked and encoded independently of the point cloud. This raises the question whether a single image might be enough, and whether the use of images improves the reconstruction at all. To investigate these questions, we construct two network variants that differ w.r.t. the number and combination of input modalities but are otherwise identical. In particular, we keep the network architecture fixed and train each variant using the same training settings and data samples. The network configuration based on stereo guidance is our default setting, referred to as IMPLICITY-stereo (or simply IMPLICITY if not stated otherwise). The first variant, IMPLICITY-mono, uses only a single ortho-image to generate the latent embedding ξ. Therefore, it cannot exploit stereo information (in the form of misalignment between orthophotos) and has to make do with image patterns and textures from a single image, with no redundancy. The second variant, IMPLICITY-0, has no access to image information. It learns the mapping from 3D points to occupancies constrained only by the shape embedding ψ, i.e., the local point distribution. Note that this configuration corresponds to the original *Convolutional Occupancy Networks* proposed in (Peng et al., 2020).

3.3 Training and Inference

At training time, we randomly sample query points $\{\mathbf{x}_i \in \mathbb{R}^3\}$ within the volume of interest and in the vicinity of the true surface (see Section 3.4). The training is supervised by the binary cross-entropy loss $\mathcal{L}$ between the predicted occupancies $\hat{o}$ and the true occupancies o at these points:

$$\mathcal{L}(\hat{o}, o) = \sum_i \big(o_i \cdot \log(\hat{o}_i) + (1 - o_i) \cdot \log(1 - \hat{o}_i)\big) \,. \qquad (2)$$

True occupancies o_i are derived from an existing city model of the training region. At inference time, we sample a regular 3D grid of query points in a hierarchical fashion, see Section 3.5.

3.4 Spatial Sampling

One challenge when training implicit neural shape models is to reach the right balance between expressiveness and generality, which boils down to sampling adequate 3D points $\mathbf{x}_i$ during training. If points were uniformly sampled in 3D space, most points would be far away from all surfaces. Consequently, the learned model would be biased towards predicting free space, as the dominant class in the absence of strong surface cues; and towards overly smooth reconstructions, since it has rarely seen surface details during training. On the other hand, if the points were exclusively sampled in the vicinity of the surface, the model would be prone to overfitting the training set, since the learning would narrowly focus on specific properties of the training area that may not generalize to other parts of the space.

In our approach, we combine uniform sampling and surface sampling, a strategy that has proven efficient for implicit neural models (Saito et al., 2019). To begin with, we uniformly sample a first set of points arbitrarily within the volume of interest. Second, we densely sample a second set of points on the true surface and perturb them with zero-mean Gaussian random noise, for our data with standard deviation $\sigma = 0.4$ m. See Section 4.1 for details. The two sets are then merged and together form the training set. In our experiments, the ratio between arbitrary points and surface points is 1:4.

3.5 Surface Extraction

To turn the implicit function f_θ into an explicit surface representation, we use a conventional raster DSM with a grid spacing of 0.25 m. Inspired by the *Multi-resolution Iso-Surface Extraction* algorithm of (Mescheder et al., 2019), we employ a hierarchical refinement scheme to extract the iso-surface $\hat{o} = 0.5$ from the occupancy volume. This approach makes it possible to recover a high-resolution DSM without having to densely sample the entire height range.

We start by discretizing the volume of interest into a regular grid of 3D points with a horizontal resolution equal to the grid spacing of the DSM and an initial vertical resolution of 16 m. Next, the occupancy of every grid point is predicted with the trained IMPLICITY model. Using the fact that in a 2.5D DSM there is exactly one transition per pixel from free to occupied space, we mark the highest occupied 3D point per (x, y)-column and the one immediately above it as active, increase the vertical resolution between the two active points by a factor 4, and predict the occupancy of the three newly generated points. Then, we again zoom in on the highest occupied point and the one immediately above it and repeat the refinement. Four iterations of this refinement lead to a final nominal resolution of 6.25 cm in the vertical direction. The highest occupied point after the last iteration is declared the DSM height $z(x, y)$. Going down to such a low nominal resolution helps to avoid aliasing artefacts on the reconstructed surface, even though it is, of course, far below the effective vertical resolution achievable with satellite images of ≈ 0.5 m GSD at nadir.

4. EXPERIMENTS

4.1 Dataset and Preprocessing

Imagery and Study Area. We evaluate our method on panchromatic satellite images acquired over Zurich, Switzerland. We have one WorldView-3 and 14 WorldView-2 images at our disposal. They were captured between 2014 and 2018, with 22 days the shortest time interval between two acquisitions. The average GSD is ≈ 0.5 m at nadir. The study area[1] covers 4 km^2 and includes widely spaced, detached residential buildings, allotments, and high commercial buildings. Moreover, it contains a stretch of the river Limmat and a forested hill. In analogy to (Stucker and Schindler, 2022), we split the area into five equally large, mutually exclusive stripes and allocate three stripes for training, one for validation, and one for testing.

Point Cloud Generation. We use a re-implementation of state-of-the-art hierarchical semi-global matching (Rothermel et al., 2012), tailored to satellite images, to generate the input point cloud $\mathcal{P}$. First, we employ the method of (Patil et al., 2019) to perform bias correction of the supplied rational polynomial coefficient (RPC) projection models. Next, we determine suitable image pairs for dense matching based on heuristics inspired by (Facciolo et al., 2017, Qin, 2019). Starting from all possible image pairs, we eliminate those whose intersection angles in object space are $<5^\circ$ or $>30^\circ$ (measured at the center of the region of interest), or whose incidence angles are $>40^\circ$ (mean of the two images). We further discard image pairs whose difference in sun angle is $>35^\circ$. To leverage the redundancy in the image set as much as possible, we use all remaining image pairs for pairwise rectification and pairwise dense matching, irrespective of differences in acquisition time, as suggested by (Krauß et al., 2019). After matching, we use the inverse RPC projection function to triangulate corresponding points per image pair, resulting in 26 stereo clouds in the same scene coordinate system, which we simply merge into a single point cloud $\mathcal{P}$.

Initial DSM Reconstruction. Besides the point cloud $\mathcal{P}$, IMPLICITY receives two ortho-rectified panchromatic satellite images as input. For the ortho-rectification, we require an initial surface estimate of the observed scene. To generate it, we fuse the point cloud $\mathcal{P}$ into a coherent multi-view raster DSM with a grid spacing of 0.25 m, by computing the cell-wise median of the n highest 3D points, where n is defined as the average number of 3D points per grid cell (Rothermel et al., 2016). Further, we adopt standard post-processing operations from aerial and satellite-based photogrammetry to denoise the DSM, remove spikes, and fill cells without a valid height with inverse distance weighted (IDW) interpolation.

Stereo Pair Selection and Rectification. Among all available image pairs with adequate stereo geometry (see above), we determine a single best pair that serves as the second input to IMPLICITY. The selection is based on three criteria, namely low intersection angle, small time difference between acquisitions (similar season), and low cloud coverage. Like (Stucker and Schindler, 2022), we ortho-rectify the two selected images with the help of the initial DSM, without ray-casting to detect occlusions. Instead, duplicate gray-values are rendered for rays that intersect the surface twice, leading to systematic patterns of repeated, photometrically inconsistent textures. Due to the small baseline between the two views, discrepancies between the ortho-images (except for illumination and atmospheric effects) primarily stem from height errors in the initial DSM rather than from viewpoint differences.

Ground Truth Occupancy. To train our method, we need to know the true occupancy of any 3D spatial location sampled within the volume of interest. Fortunately, such full 3D supervision can be readily derived from the publicly available city model of Zurich (City of Zurich, Geomatics & Surveying Department, 2018). The model has been created by the municipal surveying department in a semi-automatic manner, by fusing airborne laser scans, building and road boundaries (including bridges) from national mapping data, and roof models derived by manual stereo digitization. The height accuracy is specified as ±0.2 m for buildings and ±0.4 m for terrain.

We densely sample points on roofs, facades, and terrain of the city model. The average distance between nearest points amounts to 0.2 m for points sampled on facades and terrain. For points sampled on roofs, we increase the sampling resolution to 0.1 m to capture geometric details such as dormers with higher fidelity. Furthermore, we uniformly sample points within the volume of interest with a mean distance of ≈ 1.0 m between points. Points sampled on the surface are assigned true occupancy values of 1; points sampled in free space are assigned 0 or 1, depending on whether they lie above or below the surface.

[1] The area corresponds to ZUR1 of (Stucker and Schindler, 2022).

4.2 Implementation Details

We randomly sample training patches with a spatial dimension of 64×64 m in world coordinates from the training region. To avoid biases due to the specific topography and urban layout, we augment the data by randomly rotating the training patches by $\alpha \in \{0^\circ, 90^\circ, 180^\circ, 270^\circ\}$ and random flipping along the x and y axes. At inference time, we reconstruct large-scale scenes by applying the learned model in a sliding window.

We follow best practice and normalize the data (point cloud, ortho-images, query points) for neural network training. Every 64×64 m patch, originally given in UTM coordinates (zone 32T), is first horizontally shifted and scaled such that all point coordinates lie in $[0, 1]$. Then, all points are vertically centered to the median height and rescaled with a fixed factor. That factor is found by computing standard deviations of the heights for 20'000 random patches from the training set and averaging them (cropped to the 5th and 95th percentile for robustness). Ortho-images are normalized with the mean and standard deviation over the intensity values of all training pixels.

We have implemented ImpliCity in PyTorch and run it on a NVIDIA GeForce GTX 2080 Ti GPU. Source code and pretrained models are available at `https://github.com/prs-eth/ImpliCity`. In all experiments, we use a hidden feature dimension d of 32 for both encoders and the joint decoder, and feature plane dimensions 128×128 for the shape embedding ψ and 64×64 for the image embedding ξ. For training, we employ the ADAM optimizer with a base learning rate of $5 \cdot 10^{-5}$ (β_1=0.9, β_2=0.999), no weight decay, and a cyclical learning rate scheduler (Smith, 2017) with cycle amplitude $5 \cdot 10^{-4}$. We set the batch size to 1 but accumulate gradients for 64 training iterations before performing back-propagation. Errors at water and forest pixels are down-weighted by a factor of 0.5 when computing the loss (Eq. 2). We stop training once the DSM metrics (cf. Section 4.4) on the validation set have converged. We experimentally found that reconstruction quality improves when areas with evident temporal differences between the satellite imagery and the city model are masked out during training.

4.3 Baselines

We compare ImpliCity against the following baselines:

Initial DSM: The raster DSM generated from the input point cloud $\mathcal{P}$, representative of conventional satellite-based reconstruction (see Section 4.1 for details).

ResDepth: A learned DSM refinement approach by (Stucker and Schindler, 2022) that directly refines the initial raster DSM with a U-Net (Ronneberger et al., 2015). ResDepth-0 is trained to regress an additive height correction at every pixel. The image-guided variants ResDepth-mono and ResDepth-stereo exploit one and two ortho-images as additional input to guide the refinement.

PIFu: The Pixel-aligned Implicit Function (PIFu) method of (Saito et al., 2019), representative for deep, implicit surface reconstruction from images. We feed the initial DSM as input to the network. This baseline, denoted PIFu-0, corresponds to learned DSM filtering with the help of an implicit neural scene representation. Moreover, we train PIFu-mono and PIFu-stereo variants with one, respectively two ortho-images as additional input channels. To remain consistent with the other methods, we use a patch size of 256×256 pixels (64×64 m in scene space) rather than 512×512 as in the original PIFu.

4.4 Quality Metrics

We use the publicly available city model of Zurich (City of Zurich, Geomatics & Surveying Department, 2018) to evaluate the performance of ImpliCity. That city model is delivered in the form of 2.5D building models and a terrain surface. Therefore, we resort to 2.5D metrics commonly used for DSM evaluation. Regions where the city model differs from the images due to recent construction activities have been masked out. We render a reference DSM from the city model to measure the mean absolute error (MAE), the root mean square error (RMSE), and the median absolute error (MedAE), computed over per-pixel deviations between predicted and reference heights.[2] For a more in-depth analysis, we calculate the metrics separately for building and terrain pixels (according to the ground truth), where the building mask has been dilated by two pixels (0.5 m) to ensure that distortions along its contours are reflected in the building error. Moreover, we differentiate between general terrain and forested areas with the help of a manually created forest mask.

4.5 Results

In the following, we analyze the performance of ImpliCity and compare it to the baselines described above. We present quantitative results in Table 1 and visual examples in Figure 3.

ImpliCity-0. We start by assessing the performance of ImpliCity-0, so as to quantify the impact of implicit neural modeling in isolation, without image guidance. The DSM generated by conventional photogrammetric processing serves as the baseline. Recall, that DSM was generated by rasterizing the point cloud $\mathcal{P}$ with a cell-wise median of the highest 3D points, followed by standard denoising (cf. Section 4.1). Due to the limited resolution and inhomogeneous radiometry of satellite imagery, the resulting DSM is fairly noisy and lacks sharp features (Fig. 3, 1st row). The MAE is ≈ 3.9 m and the MedAE 1.6 m. The RMSE is almost 2× higher than the MAE because of a small number of substantial matching errors. Applying ImpliCity-0 to the point cloud $\mathcal{P}$ improves the reconstruction significantly. The MAE is lowered to ≈ 1.9 m, an improvement of >50% compared to the conventional, heuristic procedure. Similarly, the RMSE and MedAE are also reduced to 3.6 m, respectively 0.9 m. We note that vegetation is not included in the ground truth. Therefore, ImpliCity-0 learns to filter out vegetation, which naturally leads to improved quality metrics. Nonetheless, even when excluding forested areas from the evaluation, we observe a decrease in MAE of almost 43% to 1.6 m for general terrain. For buildings, the MAE drops by 25% to 2.3 m. Visually, the DSM generated with ImpliCity-0 exhibits markedly less surface noise, and built-up areas are separated into plausible individual units (Fig. 3, 5th row). Despite the limited quality and sparsity of the input point cloud, ImpliCity-0 reconstructs sharp building edges and gable roofs and even recovers buildings that are hardly visible in the conventional DSM (Fig. 3, 5th row, 2nd column). Building footprints are, however, somewhat wobbly, and visually discernible roof details are missed entirely (Fig. 3, 5th row, 3rd column).

Influence of Image Guidance. In addition to the shape code $\psi(\mathcal{P}, \mathbf{x})$, ImpliCity-mono and ImpliCity-stereo exploit a pixel-aligned image code $\xi(\mathcal{I}, \mathbf{x})$ to guide the occupancy prediction. In this way, both image-guided network variants can, on the one hand, potentially capture general cor-

[2] These widely used pixel-wise metrics do not fully characterize DSM quality: improved reconstruction of intricate geometric details may not be reflected in lower errors, see Sec. 4.5.

Reconstruction	Overall			Buildings			Terrain			Terrain w/o forested areas		
	MAE	RMSE	MedAE	MAE	RMSE	MedAE	MAE	RMSE	MedAE	MAE	RMSE	MedAE
Conventional DSM	3.89	7.03	1.59	3.02	5.02	1.47	4.29	7.78	1.65	2.79	4.69	1.43
ResDepth-0	2.21	4.20	0.98	2.84	5.34	1.10	1.93	3.56	0.94	1.50	2.71	0.84
PIFu-0	1.99	3.83	0.98	2.88	5.30	1.24	1.58	2.92	0.89	1.50	2.90	0.84
ImpliCity-0 (ours)	1.87	3.57	0.92	2.26	4.55	0.94	1.69	3.02	0.91	1.59	2.94	0.86
ResDepth-mono	1.65	3.22	0.77	2.07	4.18	0.91	1.45	2.67	0.72	1.20	2.27	0.65
PIFu-mono	1.61	3.19	0.77	2.25	4.41	1.01	1.32	2.43	0.69	1.18	2.28	0.63
ImpliCity-mono (ours)	1.58	3.03	0.73	2.00	4.03	0.80	1.39	2.43	0.69	1.23	2.24	0.62
ResDepth-stereo	1.53	2.97	0.74	1.91	3.93	0.82	1.35	2.41	0.71	1.15	2.12	0.65
PIFu-stereo	1.53	3.04	0.73	2.13	4.28	0.91	1.20	2.20	0.66	1.11	2.11	0.61
ImpliCity-stereo (ours)	1.52	2.91	0.70	1.93	3.86	0.78	1.33	2.35	0.67	1.20	2.22	0.60

Table 1. Quantitative comparison of ImpliCity with conventional DSM generation, and with learned DSM refinement based on either explicit (ResDepth) or implicit (PIFu) 2D representations (cf. Section 4.3). Methods are grouped according to their inputs: *-0:* point cloud/DSM, *-mono*: point cloud/DSM and one ortho-image, *-stereo*: point cloud/DSM and two ortho-images. All values are meters. Gray numbers are best for a given input, yellow numbers are best overall.

relations between image patterns and the underlying surface shape and, on the other hand, learn to precisely align the reprojected 3D scene geometry with 2D image discontinuities. ImpliCity-mono, leveraging a single ortho-image to generate ξ, boosts the reconstruction accuracy by a significant margin compared to ImpliCity-0. The overall MAE is decreased by 0.3 m to $\approx$1.6 m, and the MedAE by 0.2 m to 0.7 m. With a relative improvement of 18%, we observe the largest gain in accuracy for terrain. For buildings, all metrics improve by $\approx$12%. Qualitatively, ImpliCity-mono produces sharper and straighter building outlines and roof features, reconstructs buildings missed by ImpliCity-0, and partially recovers geometric details on roofs (Fig. 3, 6th row, 1st column). ImpliCity-stereo, using a second image to generate the latent code ξ, further improves the reconstruction. While the quantitative gains are rather small, the visual quality of the reconstructed 3D geometry is clearly enhanced. Buildings have crisp roof lines, and there are fewer implausible bumps on the terrain. Perhaps most striking is the recovery of fine-grained roof structures like dormers (Fig. 3, 7th row, 1st column). Our full model achieves a MAE of 1.9 m for buildings and 1.3 m for terrain.

Comparison to Learning-based Baselines. Among all methods that do not have access to monocular or stereo information, ImpliCity-0 yields the lowest reconstruction errors (Table 1, rows 2–4). Compared to ResDepth-0, it reconstructs smoother surfaces and more accurate building heights and roof features (Fig. 3, 2nd and 5th row). Quantitatively, the difference amounts to 15% in overall MAE and to 20% in MAE for buildings. On the terrain, all methods perform comparably, except that ResDepth-0 is a bit worse in forest regions. We speculate that the differences are mostly due to the network input. ImpliCity-0 receives a rich shape code ψ that encodes the local distribution of the input point cloud and potentially includes some information about the vertical point distribution, whereas ResDepth-0 has no access to such a "vertical height profile" and must infer the height entirely from 2.5D local context. PIFu-0, a generic neural model for implicit 2.5D height field reconstruction, performs slightly better than ResDepth-0. It achieves an overall MAE of 2.0 m, which is 10% lower than ResDepth-0. The gain is primarily caused by (overly) smoothed surfaces and more accurate terrain estimates in vegetated areas. Compared to ImpliCity-0, the overall MAE is 6% higher and the building MAE is 22% higher.

With additional (monocular or stereo) image information to support the reconstruction, we observe the same trend across all methods. The variants using a single image (Table 1, rows 5–7) outperform their respective counterparts without image guidance by 16–25% in terms of overall accuracy. Stereo-enabled variants (Table 1, rows 8–10) bring another improvement of 4–7% compared to the monocular versions. Notably, we find the biggest relative gain for ResDepth and the smallest one for ImpliCity. We hypothesize that, despite having co-registered latent embeddings, it may be more difficult to discover correlations between separate image codes $\xi(\mathcal{I}, \mathbf{x})$ and shape codes $\psi(\mathcal{P}, \mathbf{x})$ than to find correlations between ortho-photos and the height raster when directly stacked into a multi-channel image. In terms of absolute metrics, ImpliCity-0 and ImpliCity-mono are superior to the respective ResDepth and PIFu variants. When using stereo images as guidance, the quantitative performance of all three methods is very similar. Nevertheless, in terms of visual quality, ImpliCity-stereo is the sole method capable of recovering small roof details like dormers on single-family houses, see Figure 3.

Computational Complexity. Implicit shape models like ImpliCity have comparatively high computational cost at inference time. For every DSM patch, one must *(i)* perform one forward pass through the shape and image encoders to build the latent embeddings ψ and ξ; and *(ii)* for every query point $\mathbf{x}$ run a forward pass through the decoder to retrieve the occupancy. With our current implementation, which has not yet been tuned for speed, the occupancy decoding takes roughly 3× longer than the feature encoding, and the entire reconstruction needs $\approx$9 mins / km^2, not counting the preceding ortho-rectification of the images. In comparison, inference with ResDepth only requires a single U-Net forward pass per DSM patch, with a run time <5 sec / km^2. To put the computation times in perspective, note that even our unoptimized implementation of ImpliCity will take only $\approx$15 hours to process a city of 100 km^2 on a single machine and would not constitute a major bottleneck of the overall reconstruction pipeline.

5. CONCLUSION

We have presented ImpliCity, a method that creates DSMs from raw photogrammetric point clouds and ortho-images with the help of an implicit neural 3D scene representation. ImpliCity is able to reconstruct DSMs at city scale and, in our experiments, reduces the MAE by >60% compared to conventional stereo pipelines. In comparison with learned DSM refinement schemes, ImpliCity is particularly good at recovering minute shape details such as dormers and produces exceptionally crisp and straight building edges. Interesting future research directions include how to encode point clouds and multi-view images jointly in a single latent representation rather than separately; and extracting full 3D surfaces from the im-

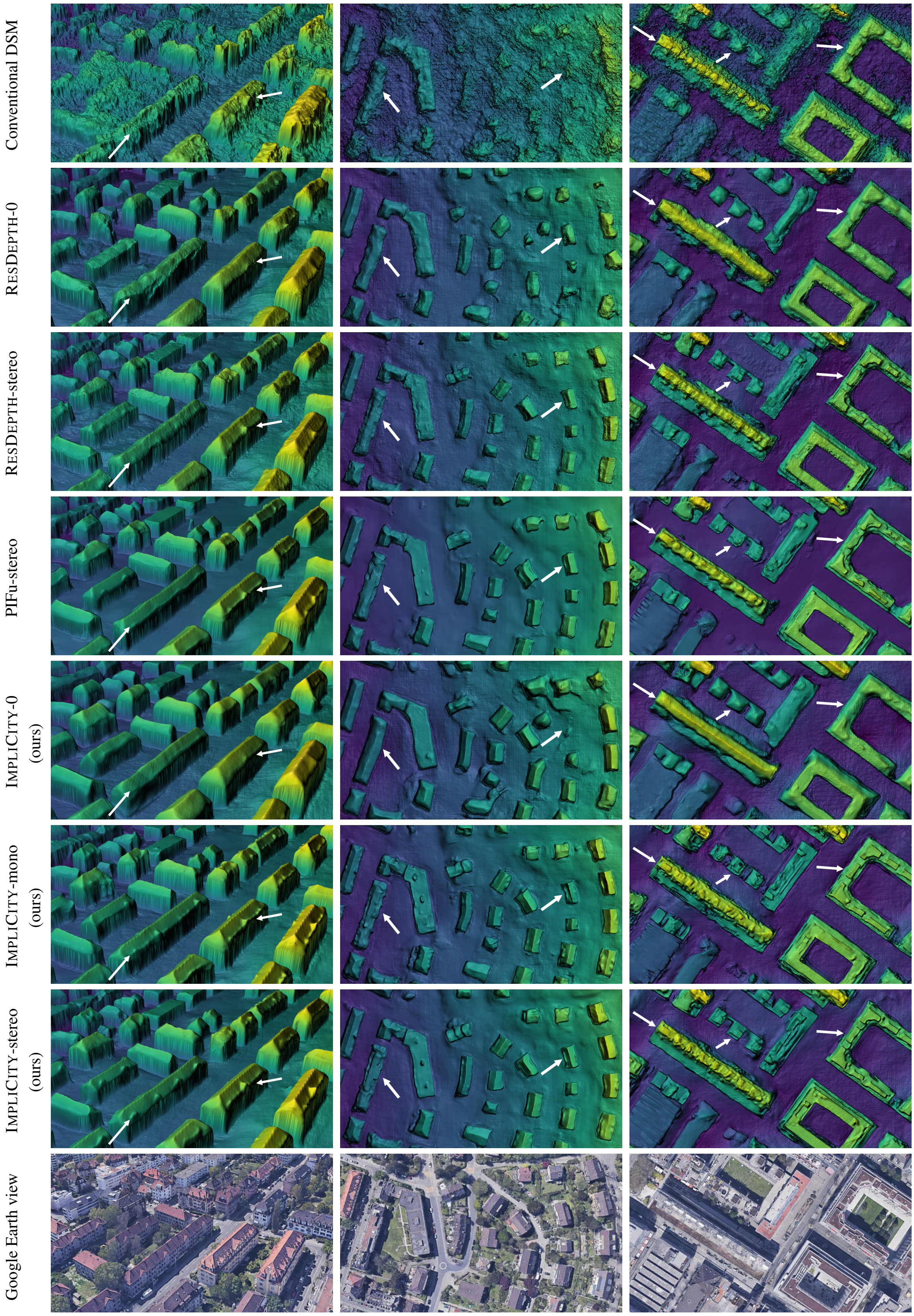

Figure 3. Visual comparison of different ImpliCity variants with selected baselines. Heights are color-coded from blue to green to yellow. All examples are from the test set.

plicit representation rather than 2.5D DSMs. ImpliCity has so far been validated under ideal machine learning conditions, with training and test regions that lie next to each other and have been observed in the same satellite images. Further work is needed to assess its ability to generalize across variations in stereo geometry, image radiometry, and geographic context. In the extended technical report (Stucker et al., 2022), we provide first, exploratory experiments for geographical generalization.

https://doi.org/10.5194/isprs-annals-V-2-2022-193-2022 | © Author(s) 2022. CC BY 4.0 License.

REFERENCES

Atzmon, M., Lipman, Y., 2020. SAL: Sign agnostic learning of shapes from raw data. *Proc. CVPR*.

Beyer, R. A., Alexandrov, O., McMichael, S., 2018. The ames stereo pipeline: NASA's open source software for deriving and processing terrain data. *Earth & Space Science*, 5(9), 537–548.

Bittner, K., Körner, M., Fraundorfer, F., Reinartz, P., 2019a. Multi-task cGAN for simultaneous spaceborne dsm refinement and roof-type classification. *Remote Sensing*, 11(11).

Bittner, K., Körner, M., Reinartz, P., 2019b. DSM building shape refinement from combined remote sensing images based on WNet-cGANs. *Proc. IGARSS*.

Bittner, K., Liebel, L., Körner, M., Reinartz, P., 2020. Long-short skip connections in deep neural networks for dsm refinement. *ISPRS Archives*, XLIII-B2-2020, 383–390.

Chabra, R., Lenssen, J. E., Ilg, E., Schmidt, T., Straub, J., Lovegrove, S., Newcombe, R., 2020. Deep local shapes: Learning local sdf priors for detailed 3d reconstruction. *Proc. ECCV*.

Chen, Z., Khademi, S., Ledoux, H., Nan, L., 2021. Reconstructing compact building models from point clouds using deep implicit fields. *arXiv preprint arXiv:2112.13142*.

Chen, Z., Zhang, H., 2019. Learning implicit fields for generative shape modeling. *Proc. CVPR*.

City of Zurich, Geomatics & Surveying Department, 2018. 3d city model. `https://www.stadt-zuerich.ch/ted/de/index/geoz/geodaten_u_plaene/3d_stadtmodell.html`.

Cournet, M., Sarrazin, E., Dumas, L., Michel, J., Guinet, J., Youssefi, D., Defonte, V., Fardet, Q., 2020. Ground truth generation and disparity estimation for optical satellite imagery. *ISPRS Archives*, XLIII-B2-2020, 127–134.

De Franchis, C., Meinhardt-Llopis, E., Michel, J., Morel, J.-M., Facciolo, G., 2014. An automatic and modular stereo pipeline for pushbroom images. *ISPRS Annals*, II-3, 49–56.

Derksen, D., Izzo, D., 2021. Shadow neural radiance fields for multi-view satellite photogrammetry. *Proc. CVPR Workshops*.

Facciolo, G., De Franchis, C., Meinhardt-Llopis, E., 2017. Automatic 3d reconstruction from multi-date satellite images. *Proc. CVPR Workshops*.

Genova, K., Cole, F., Sud, A., Sarna, A., Funkhouser, T., 2020. Local deep implicit functions for 3d shape. *Proc. CVPR*.

Genova, K., Cole, F., Vlasic, D., Sarna, A., Freeman, W. T., Funkhouser, T., 2019. Learning shape templates with structured implicit functions. *Proc. ICCV*.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. *Proc. CVPR*.

Jiang, C., Sud, A., Makadia, A., Huang, J., Nießner, M., Funkhouser, T., 2020. Local implicit grid representations for 3d scenes. *Proc. CVPR*.

Krauß, T., d'Angelo, P., Schneider, M., Gstaiger, V., 2013. The fully automatic optical processing system CATENA at DLR. *ISPRS Archives*, XL-1/W1, 177–183.

Krauß, T., d'Angelo, P., Wendt, L., 2019. Cross-track satellite stereo for 3d modelling of urban areas. *European Journal of Remote Sensing*, 52(sup2), 89–98.

Lorensen, W. E., Cline, H. E., 1987. Marching cubes: A high resolution 3d surface construction algorithm. *SIGGRAPH*.

Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S., Geiger, A., 2019. Occupancy networks: Learning 3d reconstruction in function space. *Proc. CVPR*.

Mildenhall, B., Srinivasan, P. P., Tancik, M., Barron, J. T., Ramamoorthi, R., Ng, R., 2020. NeRF: Representing scenes as neural radiance fields for view synthesis. *Proc. ECCV*.

Newell, A., Yang, K., Deng, J., 2016. Stacked hourglass networks for human pose estimation. *Proc. ECCV*.

Park, J. J., Florence, P., Straub, J., Newcombe, R., Lovegrove, S., 2019. DeepSDF: Learning continuous signed distance functions for shape representation. *Proc. CVPR*.

Patil, S., Comandur, B., Prakash, T., Kak, A. C., 2019. A new stereo benchmarking dataset for satellite images. *arXiv preprint arXiv:1907.04404*.

Peng, S., Niemeyer, M., Mescheder, L., Pollefeys, M., Geiger, A., 2020. Convolutional occupancy networks. *Proc. ECCV*.

Qi, C. R., Su, H., Mo, K., Guibas, L. J., 2017. PointNet: Deep learning on point sets for 3d classification and segmentation. *Proc. CVPR*.

Qin, R., 2016. RPC stereo processor (RSP)—a software package for digital surface model and orthophoto generation from satellite stereo imagery. *ISPRS Annals*, III-1, 77–82.

Qin, R., 2019. A critical analysis of satellite stereo pairs for digital surface model generation and a matching quality prediction model. *ISPRS Journal of Photogrammetry and Remote Sensing*, 154, 139–150.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation. *Proc. MICCAI*.

Rothermel, M., Haala, N., Fritsch, D., 2016. A median-based depthmap fusion strategy for the generation of oriented points. *ISPRS Annals*, III-3, 115–122.

Rothermel, M., Wenzel, K., Fritsch, D., Haala, N., 2012. SURE: Photogrammetric surface reconstruction from imagery. *Proc. LC3D Workshop*.

Rupnik, E., Daakir, M., Deseilligny, M. P., 2017. MicMac—a free, open-source solution for photogrammetry. *Open Geospatial Data, Software and Standards*.

Saito, S., Huang, Z., Natsume, R., Morishima, S., Kanazawa, A., Li, H., 2019. PIFu: Pixel-aligned implicit function for high-resolution clothed human digitization. *Proc. ICCV*.

Sitzmann, V., Martel, J. N. P., Bergman, A. W., Lindell, D. B., Wetzstein, G., 2020. Implicit neural representations with periodic activation functions. *Proc. NeurIPS*.

Smith, L. N., 2017. Cyclical learning rates for training neural networks. *Proc. WACV*.

Stucker, C., Ke, B., Yue, Y., Huang, S., Armeni, I., Schindler, K., 2022. ImpliCity: City Modeling from Satellite Images with Deep Implicit Occupancy Fields. *arXiv preprint arXiv:2201.09968*.

Stucker, C., Schindler, K., 2022. ResDepth: A deep residual prior for 3d reconstruction from high-resolution satellite images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 183, 560–580.

Wang, Y., Bittner, K., Zorzi, S., 2021. Machine-learned 3d building vectorization from satellite imagery. *Proc. CVPR Workshops*.

Xiangli, Y., Xu, L., Pan, X., Zhao, N., Rao, A., Theobalt, C., Dai, B., Lin, D., 2021. CityNeRF: Building NeRF at city scale. *arXiv preprint arXiv:2112.05504*.

Yang, Z., Wang, S., Manivasagam, S., Huang, Z., Ma, W.-C., Yan, X., Yumer, E., Urtasun, R., 2021. S3: Neural shape, skeleton, and skinning fields for 3d human modeling. *Proc. CVPR*.

Youssefi, D., Michel, J., Sarrazin, E., Buffe, F., Cournet, M., Delvit, J.-M., L'Helguen, C., Melet, O., Emilien, A., Bosman, J., 2020. CARS: A photogrammetry pipeline using dask graphs to construct a global 3d model. *Proc. IGARSS*.