Appendix of SynRS3D: A Synthetic Dataset for Global 3D Semantic Understanding from Monocular Remote Sensing Imagery

Jian Song[1,2], **Hongruixuan Chen**[1], **Weihao Xuan**[1,2], **Junshi Xia**[2], **Naoto Yokoya**[1,2]

[1]The University of Tokyo, Tokyo, Japan
[2]RIKEN AIP, Tokyo, Japan
song@ms.k.u-tokyo.ac.jp
https://JTRNEO.github.io/SynRS3D

A Technical Supplements

In this technical supplement, we provide detailed insights and additional results to support our main paper. Section A.1 outlines the generation process of the SynRS3D dataset, including the tools and plugins used. It also covers the licenses for these plugins. Section A.2 discusses the data sources and licenses of the existing real-world datasets utilized in our experiments. Section A.3 elaborates on the evaluation metrics for different tasks, including the proposed F_1^{HE} metric specifically designed for remote sensing height estimation tasks. Section A.4 describes the experimental setup and the selection of hyperparameters for the RS3DAda method. Section A.5 presents the ablation study results and analysis for the RS3DAda method. Section A.6 provides supplementary experimental results combining SynRS3D and real data scenarios, complementing Section 5.2 of the main paper. Section A.7 showcases the qualitative visual results of RS3DAda on various tasks. Section A.8 details the generation process and samples of building change detection annotations in SynRS3D, as well as the evaluation results of the source-only scenario on different real datasets. Section A.9 highlights the performance of models trained on the SynRS3D dataset using RS3DAda in the critical application of disaster mapping in remote sensing.

A.1 Detailed Generation Workflow of SynRS3D

The generation workflow of SynRS3D involves several key steps, from initializing sensor and sunlight parameters to generating the layout, geometry, and textures of the scene. This comprehensive process ensures that the generated SynRS3D mimics real-world remote sensing scenarios with high fidelity.

The main steps of the workflow are as follows:

- **Initialization:** Set up the sensor and sunlight parameters using uniform and normal distributions to simulate various conditions.
- **Layout Generation:** Define the grid and terrain parameters to create diverse urban and natural environments.
- **Geometry Generation:** Specify the characteristics of roads, rivers, buildings, and vegetation, ensuring realistic representations.
- **Texture Generation:** Use advanced models like GPT-4 [1] and Stable Diffusion [18] to generate realistic textures for different categories of land cover.
- **Scene Construction and Processing:** Assemble the scene with all generated components and apply textures to create visually accurate post-event and pre-event images.

38th Conference on Neural Information Processing Systems (NeurIPS 2024) Track on Datasets and Benchmarks.

Table 1: List of Blender add-ons used in the SynRS3D.

Name	Author	Version	License	URL
Realtime River Generator	specoolar	1.1	RF	https://blendermarket.com/products/river-generator
Next Street	Next Realm	2.0	RF	https://blendermarket.com/products/next-street
Objects Replacer	Georeality Design	1.06	GPL	https://blendermarket.com/products/objects-replacer/docs
Albero	Greenbaburu	0.3	RF	https://blendermarket.com/products/albero---geometry-nodes-powered-tree-generator
Hira Building Generator	HiranojiStore	0.9	RF	https://blendermarket.com/products/hira-building-generator
Procedural Building Generator	Isak Waltin	1.2.1	CC-BY 4.0	https://blendermarket.com/products/building-gen
Pro Atmo	Contrastrender	1.0	GPL	https://blendermarket.com/products/pro-atmo
Modular Buildings Creator	PH Felix	1.0	RF	https://blendermarket.com/products/modular-buildings-creator
Next Trees	Next Realm	2.0	RF	https://blendermarket.com/products/next-trees
SceneCity	Arnaud	1.9.3	RF	http://www.cgchan.com/store/scenecity
Flex Road Generator	EasyNodes	1.1.0	RF	https://www.cgtrader.com/3d-models/scripts-plugins/modelling/blender-mesh-curve-to-road
Buildify	Pavel Oliva	1.0	RF	https://paveloliva.gumroad.com/l/buildify

- **Outlier Filtering:** Filter outliers based on height maps to ensure the quality and reliability of the dataset.

The detailed algorithm for this workflow is provided in Algorithm 1. The development process of SynRS3D is based on Blender 3.4, where we utilized and modified various community add-ons to facilitate the generation of SynRS3D. A comprehensive list of all the add-ons used during our development process is presented in Table 1.

A.2 License and Data Source of Real-World Datasets

The licenses and data sources for the real-world datasets used for evaluation and training in this work are shown in Table 2. For the Potsdam, Vaihingen, GeoNRW, Nagoya, and Tokyo datasets, we used the dsm2dtm [1] algorithm to convert them to normalized Digital Surface Model (nDSM), since they only provide Digital Surface Model (DSM). We will release the processed real-world datasets upon acceptance, provided that the original datasets are allowed to be redistributed and are intended for non-commercial use.

Table 2: The data source and license of real-world height estimation datasets used in this work.

Real-World Datasets			
Types	**Datasets**	**Data Source**	**License/Conditions of Use**
Target Domain 1	Houston [23]	Data Fusion Contest 2018	Creative Commons Attribution
	JAX [12]	Data Fusion Contest 2019	Creative Commons Attribution
	OMA [12]	Data Fusion Contest 2019	Creative Commons Attribution
	GeoNRW_Urban [2]	GeoNRW	Creative Commons Attribution
	GeoNRW_Rural [2]	GeoNRW	Creative Commons Attribution
	Potsdam [19]	ISPRS	Research Purposes Only, No Redistribution
Target Domain 2	ATL [7]	Overhead Geopose Challenge	Creative Commons Attribution
	ARG [7]	Overhead Geopose Challenge	Creative Commons Attribution
	Nagoya [8]	NTT DATA Corporation and Inc. DigitalGlobe	End User License Agreement
	Tokyo [8]	NTT DATA Corporation and Inc. DigitalGlobe	End User License Agreement
	Vaihingen [19]	ISPRS	Research Purposes Only, No Redistribution

A.3 Evaluation Metrics

We utilized several metrics to ensure a comprehensive assessment of model performance when evaluating land cover mapping and height estimation tasks. In the following parts, we provide a detailed explanation and formulation of adopted metrics.

A.3.1 Land Cover Mapping

Intersection over Union (IoU) Intersection over Union (IoU) is a common evaluation metric used in image segmentation tasks. It measures the overlap between the predicted segmentation and the ground truth segmentation. The IoU for a single class is defined as:

$$\text{IoU} = \frac{|A \cap B|}{|A \cup B|}, \tag{1}$$

where A is the set of predicted pixels and B is the set of ground truth pixels.

[1] https://github.com/seedlit/dsm2dtm

Algorithm 1 Generation Workflow of SynRS3D

Initialize Parameters
$\mathcal{S} \leftarrow \{\text{azimuth} \sim U(a_1, a_2), \text{look\_angle} \sim \mathcal{N}(\mu_1, \sigma_1), \text{GSD} \sim \mathcal{N}(\mu_2, \sigma_2)\}$ # $\mathcal{S}$: Sensor parameters
$\mathcal{L} \leftarrow \{\text{elevation} \sim U(e_1, e_2), \text{intensity} \sim U(i_1, i_2), \text{color} \sim [U(c_1, c_2), U(c_1, c_2), U(c_1, c_2)]\}$ # $\mathcal{L}$: Sunlight parameters
Generate Layout
$\mathcal{G} \leftarrow \{\text{district\_num} \sim \text{randint}(d_1, d_2), \text{district\_size} \sim \text{randint}(s_1, s_2), \text{obj\_density} \sim U(o_1, o_2)\}$ # $\mathcal{G}$: Grid parameters
$\mathcal{T} \leftarrow \{\text{flat\_area} \sim U(f_1, f_2), \text{mountain\_area} \sim U(m_1, m_2), \text{sea\_area} \sim U(s_1, s_2), \text{tree\_density} \sim U(t_1, t_2)\}$ # $\mathcal{T}$: Terrain parameters
Generate Geometry
$\mathcal{R} \leftarrow \{\text{river\_num} \sim \text{randint}(r_1, r_2), \text{road\_num} \sim \text{randint}(r_3, r_4), \text{width} \sim U(w_1, w_2)\}$ # $\mathcal{R}$: Road and River parameters
$\mathcal{B} \leftarrow \{\text{height} \sim U(h_1, h_2), \text{type} \in \text{select}(\text{types}), \text{roof\_angle} \sim U(ra_1, ra_2)\}$ # $\mathcal{B}$: Building parameters
$\mathcal{V} \leftarrow \{\text{trunk} \sim \text{Sample\_Curve}(), \text{branch\_num} \sim \text{randint}(b_1, b_2), \text{leaf\_num} \sim \text{randint}(l_1, l_2)\}$ # $\mathcal{V}$: Tree parameters
Generate Textures
$\mathcal{C} \leftarrow \{\text{Rangeland}, \text{Agricultural Land}, \text{Bareland}, \text{Developed Space}, \text{Road}, \text{Roof}\}$ # $\mathcal{C}$: Texture categories
for category $\in \mathcal{C}$ **do**
    texture_prompts $\leftarrow$ GPT-4(category)
    textures[category] $\leftarrow$ Stable_Diffusion(texture_prompts)
end for
Construct Scene
$\mathcal{P}_s \leftarrow \text{create\_scene}(\mathcal{S} \cup \mathcal{L} \cup \mathcal{G} \cup \mathcal{T} \cup \mathcal{R} \cup \mathcal{B} \cup \mathcal{V})$ # $\mathcal{P}_s$: Post-event scene
$\mathcal{Q}_s \leftarrow \text{remove\_buildings}(\text{copy}(\mathcal{P}_s), U(rb_1, rb_2))$ # $\mathcal{Q}_s$: Pre-event scene
Process Scene
$\mathcal{P}_t \leftarrow \text{apply\_textures}(\mathcal{P}_s, \text{textures})$ # $\mathcal{P}_t$: Post-event scene with textures
$\mathcal{Q}_t \leftarrow \text{apply\_textures}(\mathcal{Q}_s, \text{textures})$ # $\mathcal{Q}_t$: Pre-event scene with textures
$\mathcal{P}_r \leftarrow \text{render\_rgb}(\mathcal{P}_t)$ # $\mathcal{P}_r$: Post-event RGB image
$\mathcal{Q}_r \leftarrow \text{render\_rgb}(\mathcal{Q}_t)$ # $\mathcal{Q}_r$: Pre-event RGB image
$\mathcal{P}_l \leftarrow \text{generate\_land\_cover}(\mathcal{P}_t)$ # $\mathcal{P}_l$: Post-event land cover mapping
$\mathcal{P}_h \leftarrow \text{generate\_height\_map}(\mathcal{P}_t)$ # $\mathcal{P}_h$: Post-event height map
$\mathcal{P}_b \leftarrow \text{generate\_building\_mask}(\mathcal{P}_t)$ # $\mathcal{P}_b$: Post-event building mask
$\mathcal{Q}_b \leftarrow \text{generate\_building\_mask}(\mathcal{Q}_t)$ # $\mathcal{Q}_b$: Pre-event building mask
$\mathfrak{C} \leftarrow \text{subtract\_masks}(\mathcal{P}_b, \mathcal{Q}_b)$ # $\mathfrak{C}$: Building change detection mask
Filter Outliers # Input: $\mathcal{P}_h$, H_T, H_m, H_s; Output: $\mathcal{F}_{\mathcal{P}_h}$ (Filtered height map list)
$H_T \leftarrow$ threshold value # Set the height threshold value
$H_m \leftarrow$ minimum threshold # Set the minimum proportion threshold
$H_s \leftarrow$ steepness value # Set the steepness value for the sigmoid function
$\mathcal{F}_{\mathcal{P}_h} \leftarrow \emptyset$ # Initialize the filtered height map set
for each $n \in \mathcal{P}_h$ **do**
    $a \leftarrow \text{read\_image}(n)$ # Read the height map as a numpy array
    $T_p \leftarrow \text{total\_pixels}(a)$ # Calculate the total number of pixels
    $A_t \leftarrow \text{count\_above\_threshold}(a, H_T)$ # Count the number of pixels above the threshold
    $P_c \leftarrow \frac{A_t}{T_p}$ # Calculate the proportion of pixels above the threshold
    if $P_c \geq H_m$ **then**
        $\mathcal{F}_{\mathcal{P}_h} \leftarrow \mathcal{F}_{\mathcal{P}_h} \cup \{n\}$ # If proportion is above minimum threshold, add to filtered list
    else
        $Pr \leftarrow \frac{1}{1+e^{-H_s \cdot (P_c - H_m)}}$ # Calculate the probability using a sigmoid function
        if random() $< Pr$ **then**
            $\mathcal{F}_{\mathcal{P}_h} \leftarrow \mathcal{F}_{\mathcal{P}_h} \cup \{n\}$ # Add to filtered list based on probability
        end if
    end if
end for
Output # Output: SynRS3D dataset
$\{\mathcal{F}_{\mathcal{P}_r}, \mathcal{F}_{\mathcal{Q}_r}, \mathcal{F}_{\mathcal{P}_l}, \mathcal{F}_{\mathcal{P}_h}, \mathcal{F}_{\mathfrak{C}}\}$ # $\mathcal{F}_{\mathcal{P}_r}$: Filtered post-event RGB images, $\mathcal{F}_{\mathcal{Q}_r}$: Filtered pre-event RGB images, $\mathcal{F}_{\mathcal{P}_l}$: Filtered post-event land cover mappings, $\mathcal{F}_{\mathcal{P}_h}$: Filtered post-event height maps, $\mathcal{F}_{\mathfrak{C}}$: Filtered building change detection masks

Mean Intersection over Union (mIoU) mIoU extends IoU to multiple classes by averaging the IoU values of all classes. If there are N classes, mIoU is calculated as:

$$\text{mIoU} = \frac{1}{N} \sum_{i=1}^{N} \text{IoU}_i, \tag{2}$$

where IoU_i is the IoU for class i. This metric provides a single scalar value that summarizes the segmentation performance across all classes.

A.3.2 Height Estimation

Mean Absolute Error (MAE) Mean Absolute Error (MAE) measures the average magnitude of the errors between the predicted heights and the true heights. Suppose the ground truth heights are Y and the predicted heights are $\hat{Y}$, and n is the number of samples. It is defined as:

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^{n} |Y_i - \hat{Y}_i|. \tag{3}$$

Root Mean Squared Error (RMSE) Root Mean Squared Error (RMSE) measures the square root of the average squared differences between predicted heights and actual heights. It is defined as:

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2}. \tag{4}$$

Accuracy Metric This metric, also called δ metric from early depth estimation work [9], evaluates the proportion of height predictions that fall within a certain ratio of the true heights. We use δ to represent a maxRatio map, which is calculated as follows:

$$\delta = \max\left(\frac{\hat{Y}}{Y}, \frac{Y}{\hat{Y}}\right). \tag{5}$$

Then, threshold values η are used to measure the accuracy of the height predictions, the values of η are usually $1.25, 1.25^2, 1.25^3$.

F1 Score for Height Estimation (F_1^{HE}) The F_1^{HE} score innovatively applies the F1 score, typically used in classification, to the regression task of height estimation. This metric emphasizes both precision and recall in estimating significant heights. The F_1^{HE} score balances precision and recall for height predictions above a significance threshold T (e.g., 1 meter). The maxRatio is calculated as in equation 5. True Positives (TP), False Positives (FP), and False Negatives (FN) are identified as follows:

$$TP = \sum \left((\hat{Y} > T \wedge Y > T) \wedge (\delta < \eta) \right), \tag{6}$$

$$FP = \sum \left(\hat{Y} > T \wedge Y \leq T \right), \tag{7}$$

$$FN = \sum \left(\hat{Y} \leq T \wedge Y > T \right), \tag{8}$$

where the values of η are usually $1.25, 1.25^2, 1.25^3$. Precision, Recall, and F_1^{HE} are then calculated as:

$$Precision = \frac{TP}{TP + FP}, \tag{9}$$

$$Recall = \frac{TP}{TP + FN}, \tag{10}$$

$$F_1^{HE} = 2 \times \frac{Precision \times Recall}{Precision + Recall}. \tag{11}$$

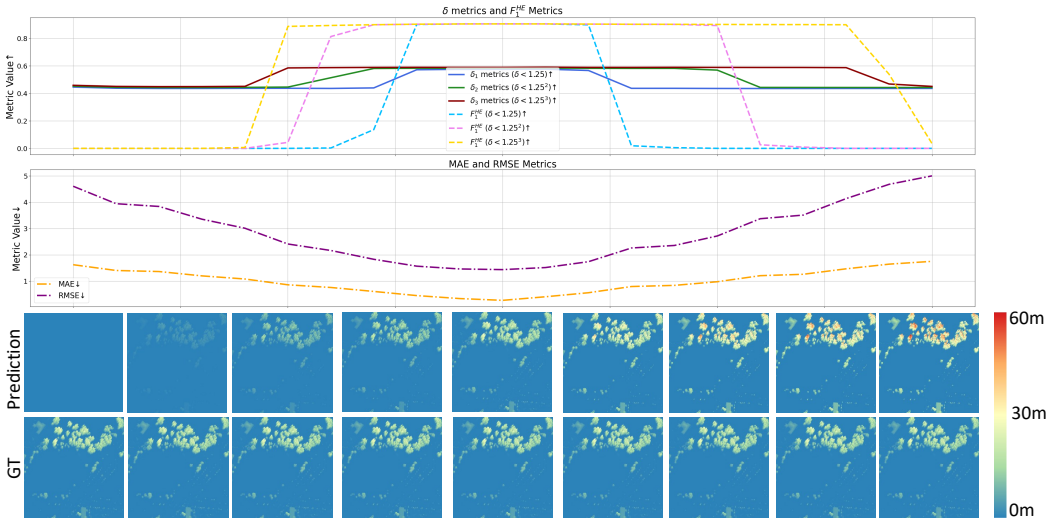

Figure 1: Comparison of proposed F_1^{HE} metric and other metrics.

Our motivation for proposing a new metric for height estimation arises from observing that existing metrics such as MAE, RMSE, and δ metrics, which are derived from depth estimation tasks, do not consider the unique characteristics of height estimation in remote sensing images. Specifically, a significant portion of the remote sensing images can be occupied by ground classes, leading to an abundance of zero height values in the ground truth. This imbalance impedes the evaluation of model performance when using traditional depth estimation metrics.

As illustrated in Figure 1, when a network predicts all values as 0 meters or predicts the height of trees and buildings as twice their ground truth (30 meters to 60 meters), metrics like MAE, RMSE, and δ still indicate highly competitive accuracy. This is not reasonable because these metrics average the correct predictions of a large number of ground pixels. However, in height estimation tasks, the accuracy of predictions for objects with height is crucial. Our proposed F_1^{HE} metric specifically addresses this issue by focusing on the accuracy of height predictions for objects higher than 1 meter. As shown, in both extreme cases, the F1 score is 0, reflecting the poor performance correctly. This metric better aligns with the objectives of the height estimation task. In practice, most images in height estimation datasets contain objects with heights exceeding 1 meter, so we skip the F_1^{HE} calculation for images that only contain ground pixels.

This comprehensive evaluation framework ensures that height estimation models are assessed on both overall error rates and the ability to accurately predict significant height values in remote sensing images.

A.4 Experimental Setting for RS3DAda

For the real-world datasets used in our experiments, we split each dataset into a 3:1 ratio for training and testing. In the RS3DAda experiments, we use random cropping of size 392 to ensure the dimensions are multiples of 14. The training batch size is set to 2, with each batch consisting of one labeled synthetic image from SynRS3D and one unlabeled image from the target domain training set.

Additionally, in RS3DAda, the teacher model is updated using Exponential Moving Average (EMA) of the student model parameters as follows:

Table 3: Settings for RS3DAda experimental hyperparameters.

Category	Parameter	Value
Statistical Image Translation [1]	Fourier Domain Adaptation (FDA)	beta_limit= 0.01
	Histogram Matching (HM)	blend_ratio= $[0.8, 1.0]$
	Pixel Distribution Adaptation (PDA)	blend_ratio= $[0.8, 1.0]$, transform_type="standard"
Strong Augmentation	ClassMix	$\lvert\mathcal{C}\rvert/2$
	ColorJitter [2]	$p = 0.8$
	GaussianBlur [3]	$p = 0.5$
Pseudo Label Generation	Land Cover Confidence Threshold (τ)	0.95
	Height Map Consistency Threshold (η)	1.55
Optimization	Optimizer	AdamW
	Encoder Learning Rate (lr)	1×10^{-6}
	Decoder Learning Rate	$10 \times lr$
	Weight Decay	5×10^{-4}
	Batch Size	2
	Iterations	$40,000$
	Warmup Steps	$1,500$
	Warmup Mode	Linear
	Decay Mode	Polynomial
	EMA (α)	0.99
Loss Function	Feature Loss Threshold (ϵ)	0.8
	Weighting Coefficient for Target Loss (λ_{target})	1
	Weighting Coefficient for Feature Loss (λ_{feat})	1

[1] `https://albumentations.ai/docs/api_reference/augmentations/domain_adaptation/`
[2] `https://albumentations.ai/docs/api_reference/augmentations/transforms/#albumentations.augmentations.transforms.ColorJitter`
[3] `https://albumentations.ai/docs/api_reference/augmentations/blur/transforms/#albumentations.augmentations.blur.transforms.GaussianBlur`

$$\theta_t \leftarrow \alpha\theta_t + (1-\alpha)\theta_s, \tag{12}$$

where θ_t represents the teacher model parameters, θ_s represents the student model parameters, and α is the EMA decay factor.

For detailed experimental parameters, please refer to Table 3.

A.5 Ablation Studies of RS3DAda

In this section, we mainly conduct ablation experiments on the three key modules of RS3DAda: 1) the ground mask, 2) height map consistency, and 3) feature constraints. Additionally, we performed ablation studies on different mixing strategies in the strong augmentation of the target domain and the setting of the number of categories in the land cover branch. The evaluation dataset for height estimation experiments is *Target Domain 2*. For land cover mapping experiments, we employed the OEM [22] dataset for evaluation.

Table 4 presents the ablation study results for the RS3DAda method. Specifically, using DINOv2 [15] and DPT [16], we find that all three modules are important for height estimation, with the ground mask and height consistency being particularly crucial. For instance, in Experiments 1 and 2, adding the ground mask reduces MAE from 6.117 to 5.652 and increases F_1^{HE} from 0.365 to 0.423. Adding height consistency in Experiment 3 further improves performance, reducing MAE to 5.253 and increasing F_1^{HE} to 0.425. The feature constraint, shown in Experiment 4, also contributes to improvements, though its impact is less significant. When all three modules are used together in Experiment 6, the best results are achieved with a MAE of 4.886, F_1^{HE} of 0.485, and mIoU of 48.23. For land cover mapping, height consistency is essential. Without it, the model relies on land cover confidence for height regression, which is often insufficient. This lack of confidence in the pseudo labels for the height branch hinders the improvement of the height estimation branch, subsequently affecting the land cover branch. These results indicate that both branches support each other, and inadequate learning in one branch negatively impacts the other.

Interestingly, with the weaker network combination of DeepLabv2 [6] and ResNet101 [10] (Experiments 7-9), the feature constraint is ineffective. This is because the ImageNet-pretrained feature extractor, trained on natural images, does not generalize well to synthetic remote sensing data, unlike DINOv2's self-supervised pretraining on diverse datasets. Aligning features with the ImageNet-pretrained extractor hinders learning from synthetic data due to the significant domain gap. This demonstrates our method's effectiveness in leveraging DINOv2's features as a constraint.

Table 4: Ablation experiments of two types of network structures with our key modules, which were introduced in the RS3DADa section. MAE and F_1^{HE} serve as evaluation metrics for the height estimation tasks, and IoU is used for the land cover mapping tasks.

#	Model	Ground Mask	Height Consistency	Feature Constraint	Height Estimation		Land Cover Mapping
				MAE ↓	F_1^{HE} ($\delta < 1.25$) ↑	mIoU ↑	
1	DPT+DINOv2	–	–	–	6.117	0.365	42.60
2	DPT+DINOv2	✓	–	–	5.652	0.423	44.05
3	DPT+DINOv2	✓	✓	–	5.253	0.425	44.75
4	DPT+DINOv2	✓	–	✓	5.578	0.439	42.93
5	DPT+DINOv2	–	✓	✓	5.384	0.461	46.67
6	DPT+DINOv2	✓	✓	✓	**4.886**	**0.485**	**48.23**
7	DLv2+R101	–	–	–	7.419	0.318	17.42
8	DLv2+R101	✓	✓	✓	6.959	0.316	18.89
9	DLv2+R101	✓	✓	–	**6.708**	**0.352**	**22.55**

Table 5: Comparison of mixing strategies and number of classes.

Mix Strategy / #Class	Height Estimation		Land Cover Mapping
	MAE ↓	F_1^{HE} ($\delta < 1.25$) ↑	mIoU ↑
Mix Strategy			
CutMix [25]	4.966	0.475	47.34
ClassMix [14]	**4.886**	**0.485**	**48.23**
#Classes			
3	5.136	0.425	-
8	**4.886**	**0.485**	-

We also explored the impact of two different mix strategies and the number of land cover classes on the RS3DAda method. As shown in Tab. 5, ClassMix has a slight advantage over CutMix in both tasks. Regarding the number of land cover classes, we found that using all 8 land cover classes outperforms using only 3 classes (ground, tree, building). This improvement is likely because land cover mapping, being a segmentation task, benefits from a more detailed and discrete representation of features. In contrast, height estimation, which is a regression task, relies on continuous features. By having a finer label space in the classification branch, we can better align the segmentation and regression tasks, reducing the discrepancy between them.

A.6 Additional Height Estimation Results in Combining SynRS3D and Real Data Scenarios

In the Section 5.2 of the main paper, we present height estimation results for three datasets. Here, we provide the remaining results for seven additional datasets. These results further demonstrate the efficacy of combining SynRS3D with real data across different environments for fine-tuning and joint training. Figures 2 and 3 showcase the performance across these additional datasets, following the same evaluation methodology as described in Section 5.2 of the main paper. These extended results support the main paper's conclusions, demonstrating that both fine-tuning on real data after pre-training on SynRS3D (FT) and joint training with SynRS3D and real data (JT) significantly enhance model performance, especially when real data is limited. This underscores the importance of SynRS3D in complementing existing datasets and boosting model performance.

A.7 Qualitative Results of RS3DAda

Figure 4 shows the qualitative results for the height estimation task. We can observe that the height predictions from the RS3DAda model are closer to the ground truth and have more complete edges. In contrast, the source-only model tends to overestimate height values and produces more incomplete edges. Although the model trained on *Target Domain 1* uses real data, it struggles to generalize to *Target Domain 2* due to its training data being limited to commonly available public datasets from European and American regions, which are unbalanced. As shown, its predicted heights are often underestimated. Figure 5 presents the qualitative results for the land cover mapping task. The RS3DAda model demonstrates exceptional performance in categories such as agricultural land, rangeland, and bare land, which aligns with our quantitative experimental results. However, it has

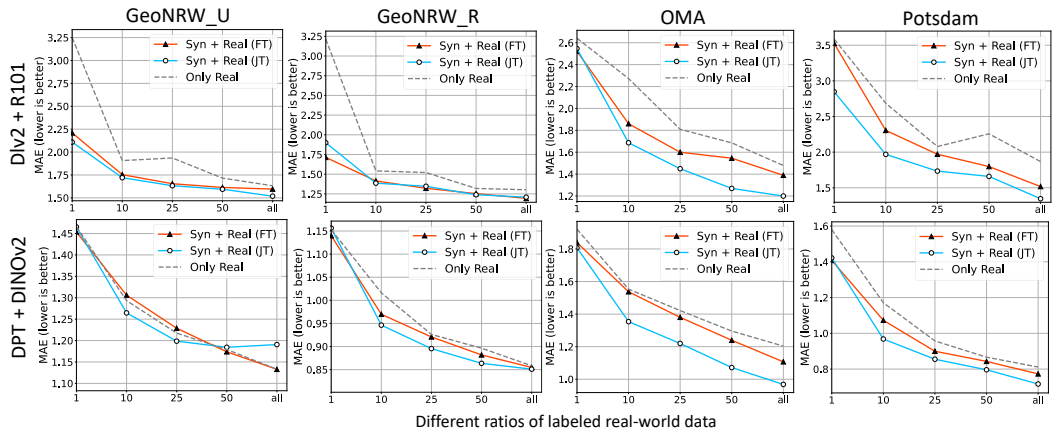

Figure 2: Additional performance evaluation on *Target Domain 1* datasets of combining SynRS3D with real data on height estimation task. FT: fine-tuning on real data after pre-training on SynRS3D, JT: joint training with SynRS3D and real data.

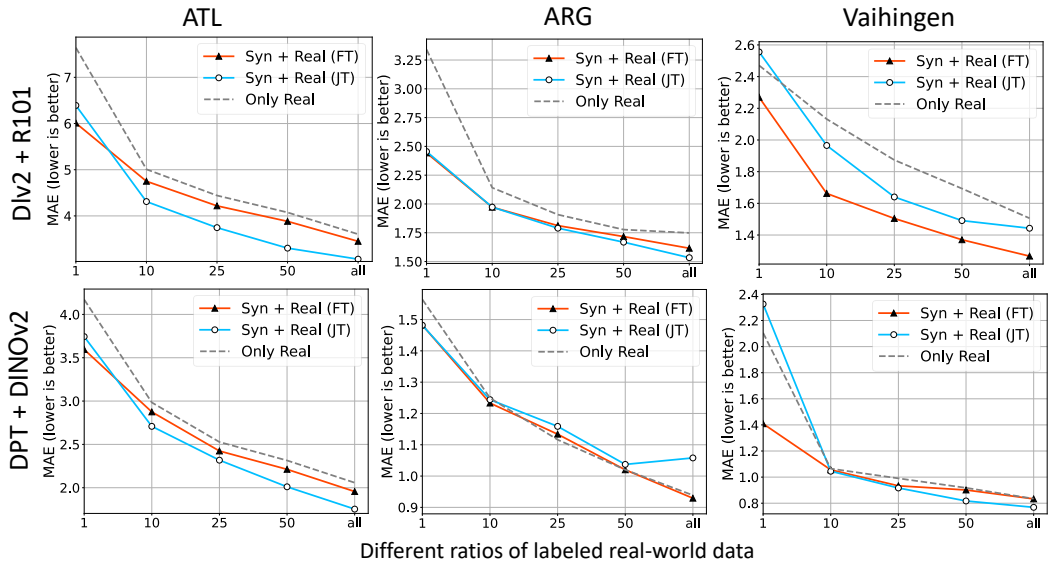

Figure 3: Additional performance evaluation on *Target Domain 2* datasets of combining SynRS3D with real data on height estimation task. FT: fine-tuning on real data after pre-training on SynRS3D, JT: joint training with SynRS3D and real data.

some limitations in categories like roads and developed space, indicating that there is still significant room for improvement in domain adaptation research for the SynRS3D dataset in the area of land cover mapping. This marks the first time in the field of remote sensing that synthetic data alone can achieve a high level of visual interpretation consistency with the ground truth. We hope that the RS3DAda method and the SynRS3D dataset can serve as benchmarks to further advance research in this direction. Figure 6 shows additional 3D reconstruction results in developing countries. These results are derived from using models trained on SynRS3D with RS3DAda to infer monocular satellite image tiles from Bing Satellite[2] and HereWeGo Satellite[3]. These 3D reconstruction areas cover between 3.2 square kilometers and 12.85 square kilometers, with a ground sample distance (GSD) of 0.35 meters.

[2] `https://www.bing.com/maps`

[3] `https://wego.here.com`

Figure 4: Qualitative results of height estimation task on *Target Domain 2* using the RS3DAda model, the source-only model, and the model trained on *Target Domain 1*. Satellite RGB images form Tokyo and Nagoya: © 2018 NTT DATA Corporation and Inc. DigitalGlobe.

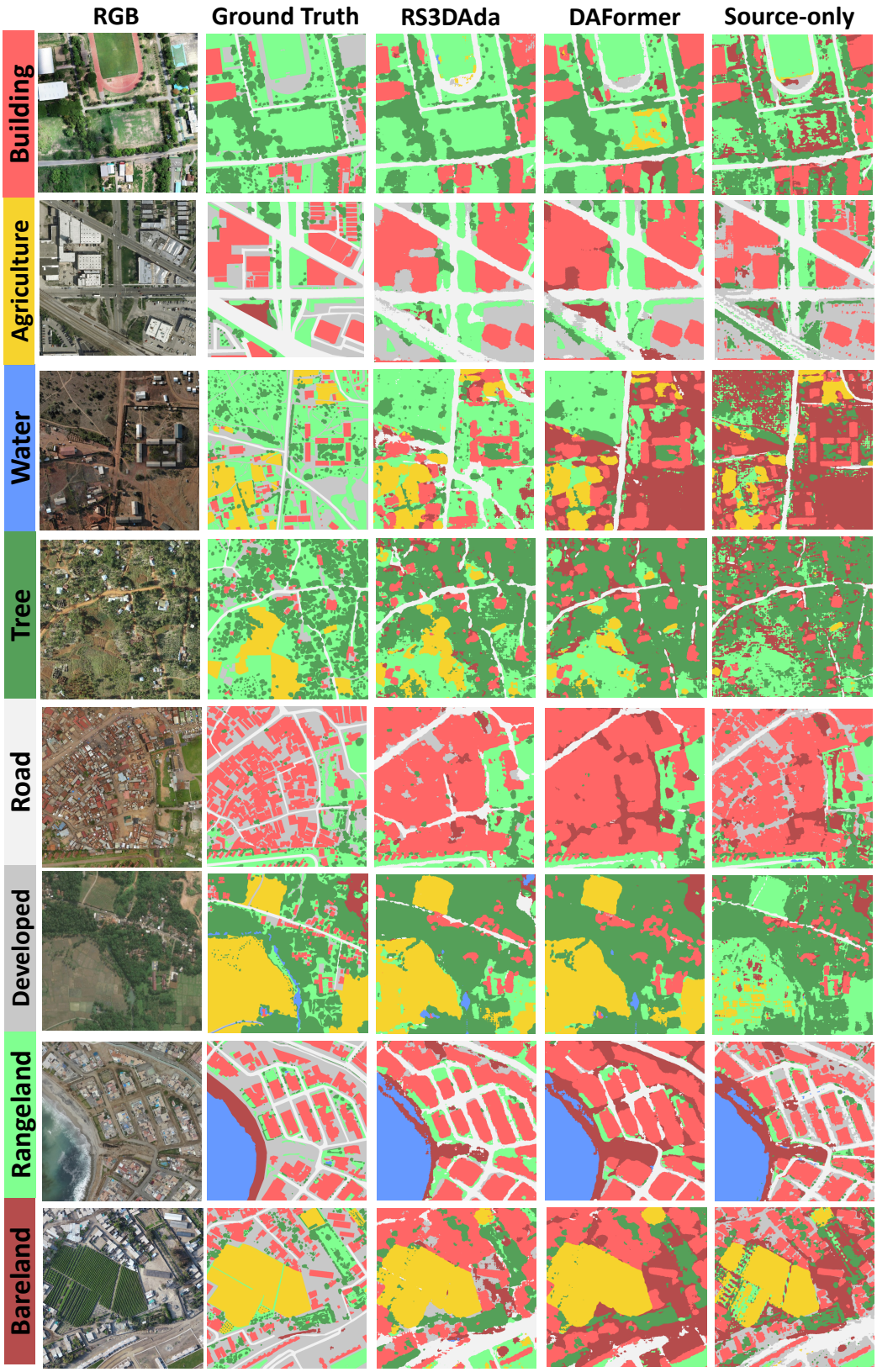

Figure 5: Qualitative results of land cover mapping task on OEM dataset using the RS3DAda model, the source-only model, and DAFormer.

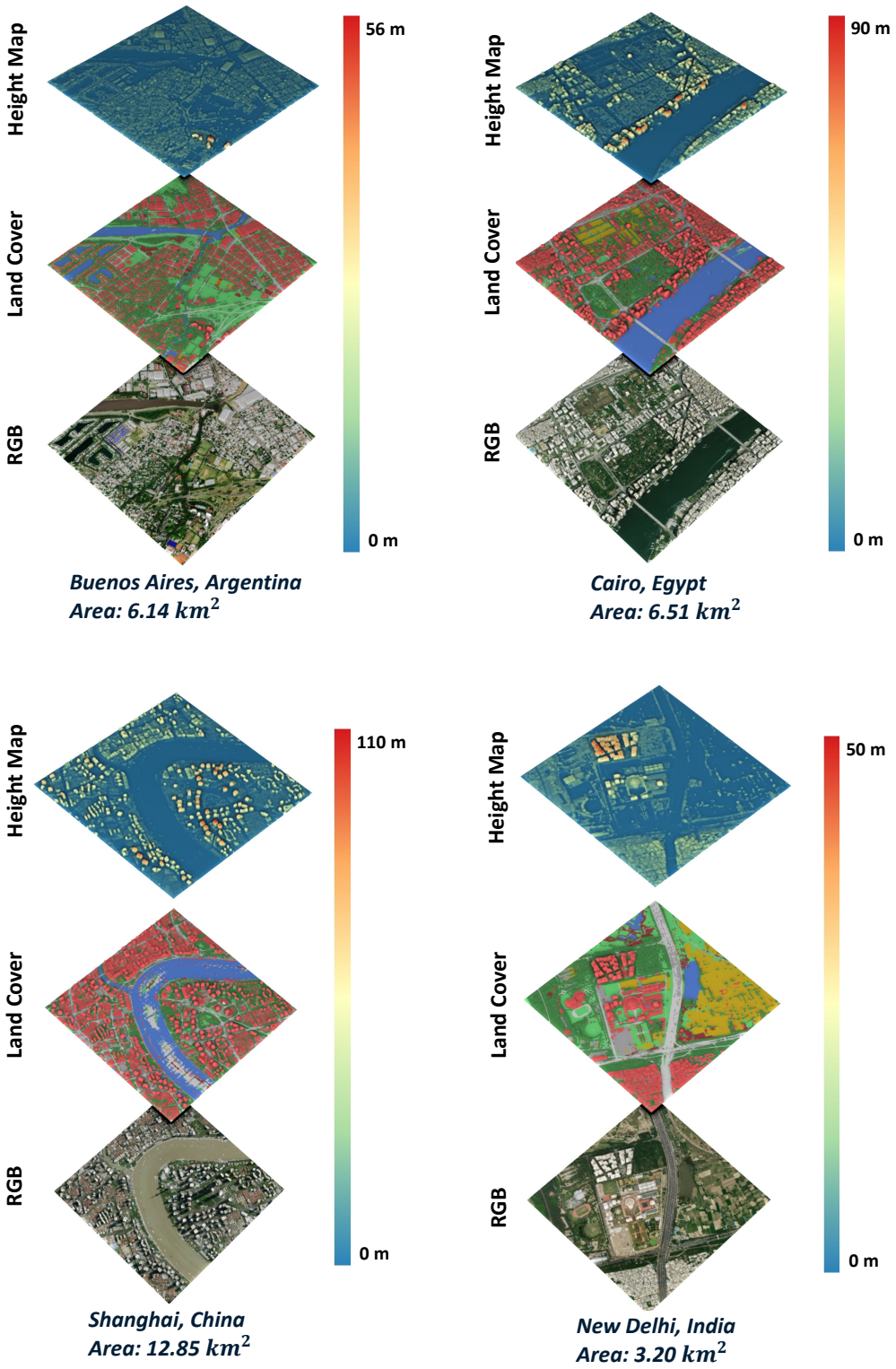

Figure 6: 3D visualization outcomes from real-world monocular RS images, which uses the model trained on SynRS3D dataset with proposed RS3DAda method. RGB satellite images of Buenos Aires and New Delhi: © HERE WeGo Satellite. RGB satellite images of Cairo and Shanghai: © Being Satellite.

A.8 Building Change Detection

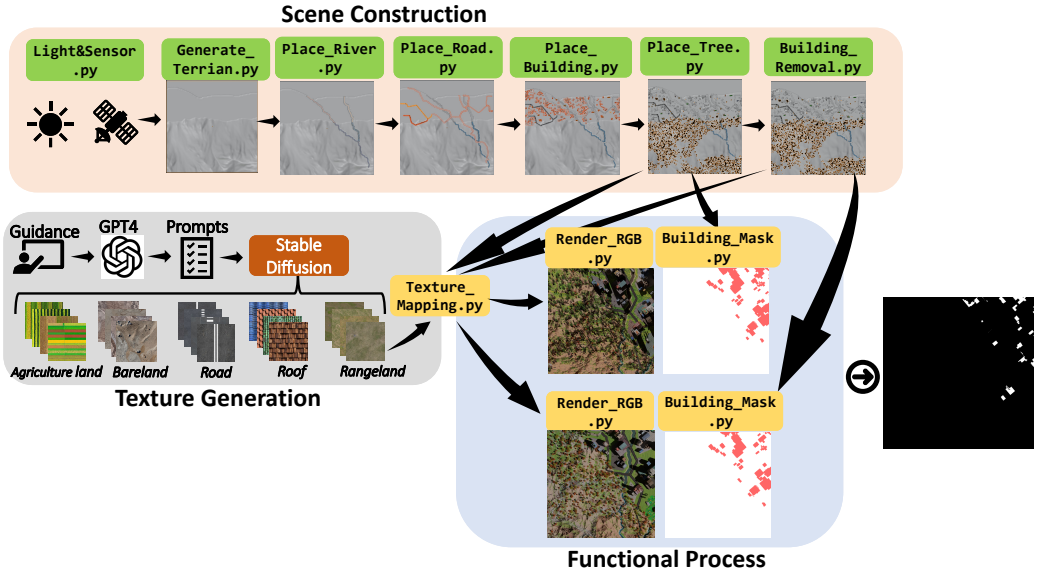

Figure 7: Generation workflow of building change detection mask.

SynRS3D provides 8-class land cover mapping annotations, accurate height maps, and binary masks specifically designed for RS building change detection tasks. The image and mask generation process is illustrated in Figure 7. For the synthesized scenes, an additional step is included: a certain proportion of buildings are randomly removed, and all geometries in the scene are retextured. Subsequently, post-event and pre-event RGB images, along with building masks, are rendered. By subtracting the two masks, the final building change mask is obtained. Figure 8 exhibits examples of change detection in six styles within SynRS3D.

To validate the effectiveness of SynRS3D in change detection tasks, we conducted experiments in a source-only scenario, where models were trained only on synthetic data and tested directly on real-world datasets. We compared our results with the models trained on two other advanced synthetic datasets, SMARS [17] and SyntheWorld [20], that include labels for the RS building change detection task. For real-world datasets, we used commonly utilized datasets in change detection tasks: WHU-CD [11], LEVIR-CD+ [4], and SECOND [24].

The WHU-CD dataset, a subset of the WHU Building dataset, focuses on building change detection with aerial images from Christchurch, New Zealand, captured in April 2012 and 2016 at 0.3 meters/pixel resolution. Covering 20.5 km^2, the dataset documents significant urban development, with buildings increasing from 12,796 to 16,077 over four years. LEVIR-CD+ is an advanced building CD dataset comprising 985 pairs of high-resolution (0.5 meters/pixel) images, documenting changes over 5 to 14 years and featuring various building types. It includes 31,333 instances of building changes, making it a valuable benchmark for CD methodologies. The SECOND dataset consists of 4,662 pairs of 512×512 aerial images (0.5-3 meters/pixel) annotated for land cover change detection in cities like Hangzhou, Chengdu, and Shanghai, but in our experiments, we only use its building change mask. These datasets were split into training and testing sets in a 3:1 ratio, with a training size of 256×256 pixels.

We employed four change detection frameworks for evaluating SMAR, SyntheWorld, and SynRS3D, including the CNN-based DTCDSCN [13], the transformer-based ChangeFormer [3], and the current state-of-the-art Mamba-based method, ChangeMamba [5]. Notably, due to our empirical findings of the strong potential of DINOV2 [15] pre-trained networks on synthetic data in both land cover mapping and change detection tasks, we implemented a framework combining the DINOV2 encoder with the ChangeMamba decoder for change detection on synthetic data, which we named DINO-Mamba. For synthetic datasets, we use a batch size of 2, and for real data, we use a batch size of 16. The optimizer used is AdamW, with a learning rate of 1e-5 for DinoMamba, while all other methods

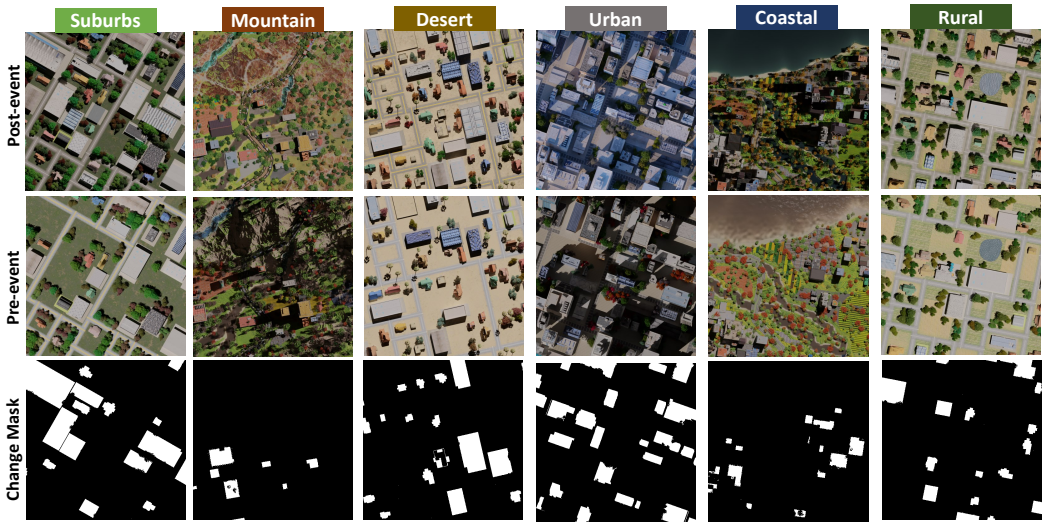

Figure 8: Examples of building change detection task in SynRS3D.

use a learning rate of 1e-4. All models are trained for 40,000 iterations on a single Tesla A100. The evaluation metrics used are IoU and F1.

Table 6: Peformance evaluation of building change detection task on WHU-CD [11] dataset.

Train on	DTCDSCN [13]		ChangeFormer [3]		ChangeMamba [5]		DinoMamba	
	IoU	F1	IoU	F1	IoU	F1	IoU	F1
SMARS [17]	26.84	42.55	18.67	31.88	42.50	59.63	48.11	64.87
SyntheWorld [20]	30.17	46.53	**41.73**	**58.87**	47.26	64.10	54.20	70.14
SynRS3D	**33.09**	**49.84**	35.00	51.94	**52.94**	**69.08**	**61.60**	**76.00**
Real	58.31	73.67	79.98	88.88	88.44	93.87	87.57	93.38

Table 7: Peformance evaluation of building change detection task on LEVIR-CD+ [4] dataset.

Train on	DTCDSCN [13]		ChangeFormer [3]		ChangeMamba [5]		DinoMamba	
	IoU	F1	IoU	F1	IoU	F1	IoU	F1
SMARS [17]	11.70	21.53	15.67	27.58	27.50	42.50	30.85	47.31
SyntheWorld [20]	21.16	35.28	23.31	38.12	28.28	44.30	48.78	65.46
SynRS3D	**25.82**	**41.30**	**23.33**	**38.14**	**30.39**	**46.78**	**49.63**	**66.23**
Real	63.44	77.63	67.48	80.58	77.39	87.25	74.12	85.14

Table 8: Peformance evaluation of building change detection task on the SECOND [24] dataset.

Train on	DTCDSCN [13]		ChangeFormer [3]		ChangeMamba [5]		DinoMamba	
	IoU	F1	IoU	F1	IoU	F1	IoU	F1
SMARS [17]	17.26	29.88	23.30	38.09	29.85	46.15	35.20	51.07
SyntheWorld [20]	21.00	35.07	26.44	42.06	27.23	43.02	37.61	54.71
SynRS3D	**33.52**	**50.32**	**31.36**	**47.90**	**38.88**	**56.02**	**39.18**	**56.33**
Real	58.78	74.04	60.08	75.06	67.61	80.68	67.65	80.71

Tables 6, 7, 8 present our experimental results, showing that the combination of SynRS3D and DINOMamba achieved F1 scores of 76.00, 66.23, and 56.33 on WHU, LEVIR-CD+, and SECOND respectively. Although there is still a gap compared to the Oracle model trained on real-world data, our dataset significantly boosts models' performances compared with the other two synthetic datasets. We have established a benchmark based on SynRS3D and advanced change detection networks, hoping to further promote the development of RS change detection using synthetic data.

A.9 Disaster Mapping Study Cases

The models trained on the SynRS3D dataset using the RS3DAda method can be utilized for various remote sensing downstream applications. We explored their potential in disaster mapping applications.

In February 2023, a devastating earthquake struck southeastern Turkey, primarily affecting the Kahramanmaraş region. This earthquake, with a magnitude of 7.8, caused widespread destruction, resulting in over 45,000 deaths, thousands of injuries, and massive displacement of residents. The economic losses were estimated to be in the billions of dollars. Rescue operations were carried out by both national and international teams, working tirelessly to save lives and provide aid to the affected population. Similarly, in August 2023, Hawaii experienced severe wildfires, particularly affecting the island of Maui. These wildfires, exacerbated by dry conditions and strong winds, led to extensive destruction of homes, infrastructure, and natural landscapes. The fires caused significant economic losses, displacing many residents and leading to casualties. The coordinated efforts of local authorities and fire departments, along with support from federal agencies, were crucial in controlling the fires and assisting those affected.

To assess the impact of these disasters, we used the height estimation branch of RS3DAda to infer pre- and post-event remote sensing images. By simply subtracting the predicted height maps of the post-event from the pre-event, we obtained a Height Difference map. This map was filtered using a threshold: 3 meters for the earthquake example (indicating that buildings severely damaged in the earthquake would collapse, resulting in a significant height reduction) and 1 meter for the wildfire example (assuming that changes exceeding 1 meter indicate damage in the fire). Figure 10 presents the study case for the Turkey earthquake, and Figure 9 shows the study case for the Hawaii wildfires.

This simple method allowed us to roughly delineate the affected areas and assess the damage severity based on height differences. Although not entirely precise, this approach represents a significant success in applying models trained solely on synthetic data to real-world scenarios. We believe in the potential of RS3DAda and SynRS3D in this research domain and look forward to more applications and studies in the future.

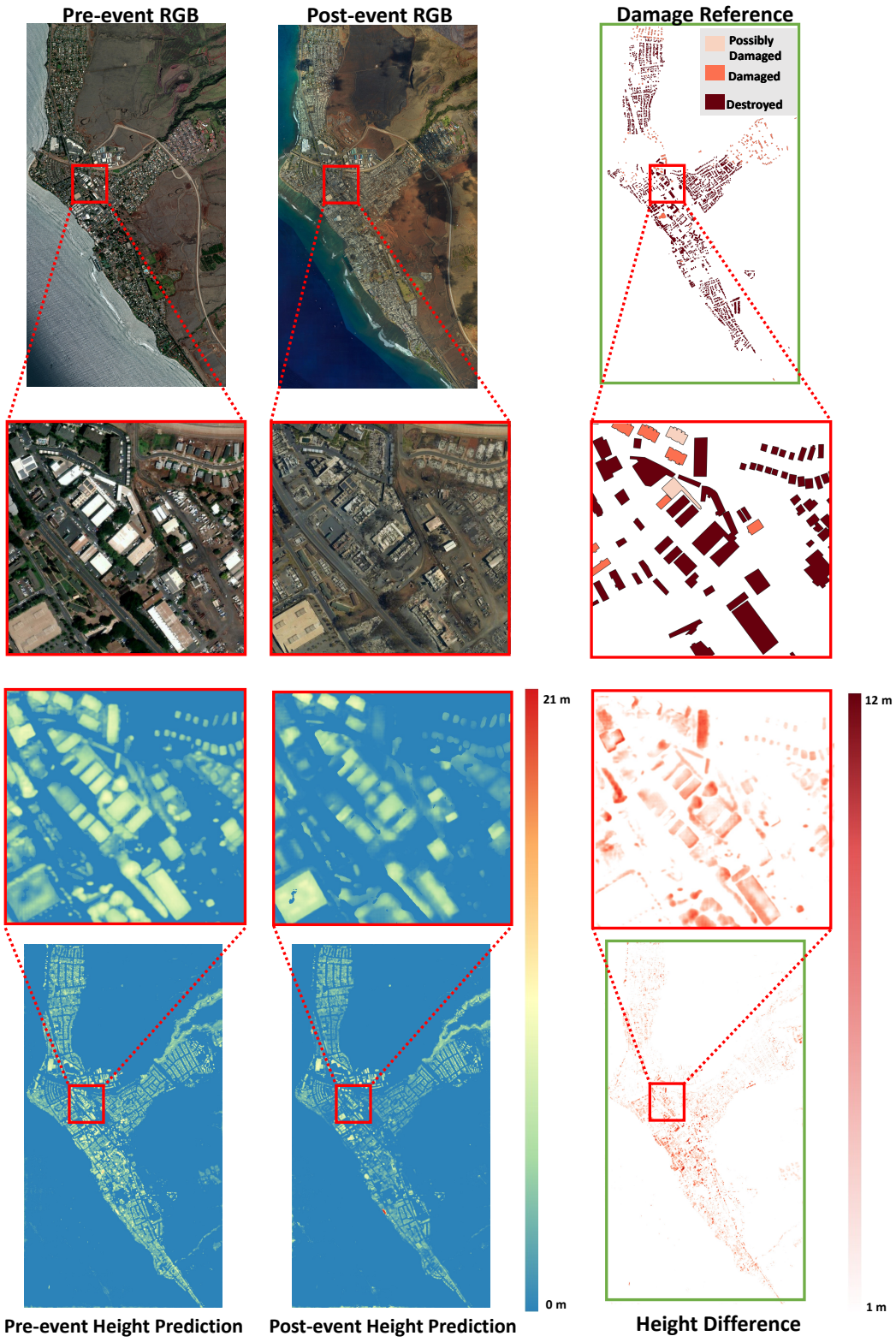

Figure 9: Study case of 2023 Hawaii-Maui wildfire. RGB satellite images of pre-event: © Being Satellite. RGB satellite images of post-event: © Google Satellite.

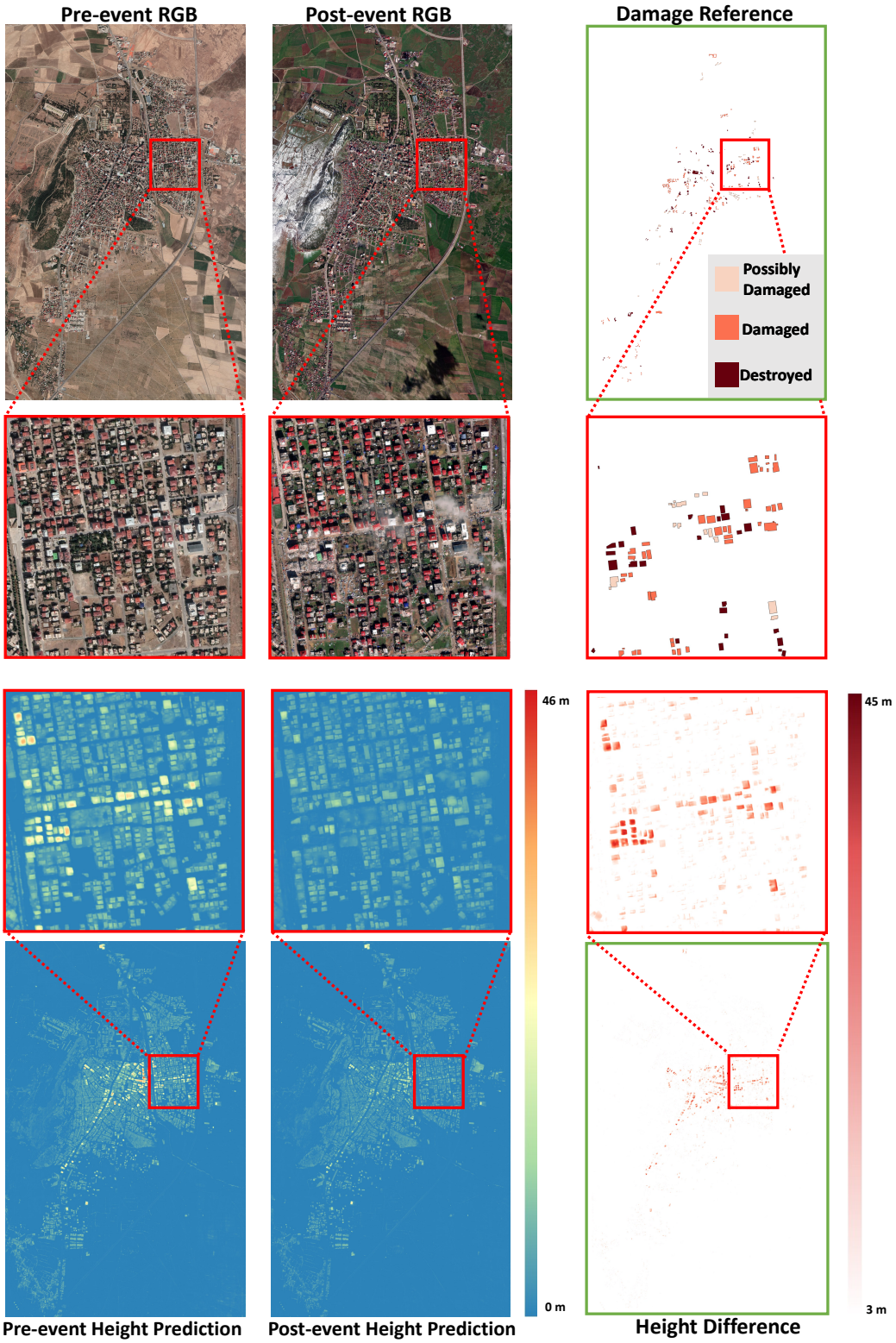

Figure 10: Study case of 2023 Turkey–Syria earthquakes. RGB satellite images: © 2023 CNES/Airbus, Maxar Technologies.

B Datasheet

B.1 Motivation

1. For what purpose was the dataset created? Was there a specific task in mind? Was there a specific gap that needed to be filled? Please provide a description.

A1: The dataset was created to enable global 3D semantic understanding from single-view high-resolution remote sensing imagery, addressing the challenges of high annotation costs, data collection, and geographically restricted data availability.

2. Who created the dataset (e.g., which team, research group) and on behalf of which entity (e.g., company, institution, organization)?

A2: SynRS3D is created by the first author and its affiliations.

3. Who funded the creation of the dataset? If there is an associated grant, please provide the name of the grantor and the grant name and number.

A3: This work was supported in part by JST FOREST Program Grant Number JPMJFR206S; Microsoft Research Asia; JSPS KAKENHI Grant Number 24KJ0652; the Next Generation AI Research Center of The University of Tokyo; the Japan Science and Technology Agency SPRING Program (JST SPRING) Grant Number JPMJSP2108; and RIKEN Junior Research Associate (JRA) Program.

B.2 Composition

1. What do the instances that comprise the dataset represent (e.g., documents, photos, people, countries)? Are there multiple types of instances (e.g., movies, users, and ratings; people and interactions between them; nodes and edges)? Please provide a description.

A1: The instances are high-resolution synthetic RGB images representing diverse geographic environments and associated annotations for height estimation, land cover mapping, and building change detection. Land cover has 8 types: bareland, rangeland, developed space, road, tree, water, agricultural land, and building. The height range is from 0m to 409m.

2. How many instances are there in total (of each type, if appropriate)?

A2: 69,667 high-resolution RGB images. Specifically, it includes 69,667 pre-event images, 69,667 post-event images, corresponding height maps, land cover annotations, and building change detection masks.

3. Does the dataset contain all possible instances or is it a sample (not necessarily random) of instances from a larger set? If the dataset is a sample, then what is the larger set? Is the sample representative of the larger set (e.g., geographic coverage)? If so, please describe how this representativeness was validated/verified. If it is not representative of the larger set, please describe why not (e.g., to cover a more diverse range of instances, because instances were withheld or unavailable).

A3: SynRS3d aims to be representative by covering six different city styles worldwide. After the initial SynRS3D dataset generation, images with anomalous height distributions are filtered out. The final version of SynRS3D was constructed using the following prior knowledge [21]: backward regions (low buildings) cover about 12% of the world's areas, emerging regions (mid buildings) cover about 70%, and developed regions (tall buildings) cover about 18%. This step ensures that the final version of SynRS3D closely aligns with real-world height distributions. The specific filtering algorithm detail can be found in Algorithm 1.

4. What data does each instance consist of? "Raw" data (e.g., unprocessed text or images) or features? In either case, please provide a description.

A4: Each instance consists of high-resolution synthetic RGB images captured before and after the event, land cover labels, accurate height maps, and building change masks. The corresponding instances are organized into folders: `gt_cd_mask`, `gt_nDSM`, `gt_ss_mask`, `opt`, and `pre_opt`. Each folder contains the respective data, and each corresponding image within these folders shares the same name.

5. Is there a label or target associated with each instance? If so, please provide a description.

A5: Yes, labels include land cover mapping, height map, and building change mask. Land cover has 8 types: bareland, rangeland, developed space, road, tree, water, agricultural land, and building. The height range is from 0m to 409m.

6. Is any information missing from individual instances? If so, please provide a description, explaining why this information is missing (e.g., because it was unavailable). This does not include intentionally removed information, but might include, e.g., redacted text.

A6: Yes, while the current SynRS3D dataset already contains sufficient information, expanding each instance to include simulated SAR images would allow the dataset to be extended to various remote sensing multimodal tasks. However, successful simulation of SAR images is currently beyond our capabilities. This will be our future work.

7. Are relationships between individual instances made explicit (e.g., users' movie ratings, social network links)? If so, please describe how these relationships are made explicit.

A7: Our dataset does not include direct relationships between individual instances, and each instance is named numerically. However, the dataset is organized into multiple batches or packages, and the relationships between these batches can be inferred from their naming conventions. Each package is named using the format "xx_yy_zz", where "xx" represents the different layouts, "yy" denotes the ground sampling distance (GSD), and "zz" indicates whether the height distribution is low, medium, or high.

8. Are there recommended data splits (e.g., training, development/validation, testing)? If so, please provide a description of these splits, explaining the rationale behind them.

A8: As a synthetic dataset, the primary concern is evaluating the performance of models trained on this dataset when applied to real datasets. Therefore, we did not create training and testing splits for SynRS3D.

9. Are there any errors, sources of noise, or redundancies in the dataset? If so, please provide a description.

A9: No.

10. Is the dataset self-contained, or does it link to or otherwise rely on external resources (e.g., websites, tweets, other datasets)? If it links to or relies on external resources, a) are there guarantees that they will exist, and remain constant, over time; b) are there official archival versions of the complete dataset (i.e., including the external resources as they existed at the time the dataset was created); c) are there any restrictions (e.g., licenses, fees) associated with any of the external resources that might apply to a dataset consumer? Please provide descriptions of all external resources and any restrictions associated with them, as well as links or other access points, as appropriate.

A10: The dataset is self-contained.

11. Does the dataset contain data that might be considered confidential (e.g., data that is protected by legal privilege or by doctor–patient confidentiality, data that includes the content of individuals' non-public communications)? If so, please provide a description.

A11: No.

12. Does the dataset contain data that, if viewed directly, might be offensive, insulting, threatening, or might otherwise cause anxiety? If so, please describe why.

A12: No.

13. Does the dataset identify any subpopulations (e.g., by age, gender)? If so, please describe how these subpopulations are identified and provide a description of their respective distributions within the dataset.

A13: No.

14. Is it possible to identify individuals (i.e., one or more natural persons), either directly or indirectly (i.e., in combination with other data) from the dataset? If so, please describe how.

A14: No.

15. Does the dataset contain data that might be considered sensitive in any way (e.g., data that reveals race or ethnic origins, sexual orientations, religious beliefs, political opinions or union memberships, or locations; financial or health data; biometric or genetic data; forms of government identification, such as social security numbers; criminal history)? If so, please provide a description.

A15: No.

B.3 Collection Process

1. How was the data associated with each instance acquired? Was the data directly observable (e.g., raw text, movie ratings), reported by subjects (e.g., survey responses), or indirectly inferred/derived from other data (e.g., part-of-speech tags, model-based guesses for age or language)? If the data was reported by subjects or indirectly inferred/derived from other data, was the data validated/verified? If so, please describe how.

A1: SynRS3D was generated using procedural modeling techniques based on Blender[4] and Python[5]. The RGB images were obtained using Blender's built-in orthographic camera and Cycles rendering engine. All distances and position information within the 3D software are known, so the labels were accurately obtained using Blender's compositor node. All images and labels are saved in TIFF format. The RGB images are three-channel uint8 format and can be viewed using various interactive image viewers (e.g., Windows Photos[6] or IrfanView[7]) or Python. Land cover annotations are single-channel uint8 format with values ranging from 1 to 8. Below is the label map and the colormap used for visualization in this work. Height map annotations are single-channel float32 format, and we recommend using more professional software such as QGIS[8] for visualization. The building change masks are single-channel images consisting of values 0 and 255.

```
colormap = {
    1: [181, 76, 76],       % Bareland
    2: [128, 255, 144],     % Grass
    3: [200, 200, 200],     % Developed space
    4: [242, 242, 242],     % Road
    5: [85, 160, 89],       % Tree
    6: [102, 153, 255],     % Water
    7: [246, 211, 45],      % Agriculture land
    8: [255, 102, 102]      % Buildings
}
```

2. What mechanisms or procedures were used to collect the data (e.g., hardware apparatuses or sensors, manual human curation, software programs, software APIs)? How were these mechanisms or procedures validated?

A2: Please refer to Section A.1.

3. If the dataset is a sample from a larger set, what was the sampling strategy (e.g., deterministic, probabilistic with specific sampling probabilities)?

A3: Please refer to Section A.1.

4. Who was involved in the data collection process (e.g., students, crowdworkers, contractors) and how were they compensated (e.g., how much were crowdworkers paid)?

A4: The first author of this paper.

[4] https://www.blender.org/

[5] https://www.python.org/

[6] https://support.microsoft.com/en-us/help/4026249/windows-10-photos

[7] https://www.irfanview.com/

[8] https://qgis.org/en/site/

5. Over what timeframe was the data collected? Does this timeframe match the creation timeframe of the data associated with the instances (e.g., recent crawl of old news articles)? If not, please describe the timeframe in which the data associated with the instances was created.

A5: Each instance took an average of 5 minutes to generate using a Tesla A100 GPU. We used 30-40 GPUs in parallel, and the total time taken was approximately one week.

B.4 Preprocessing/Cleaning/Labeling

1. Was any preprocessing/cleaning/labeling of the data done (e.g., discretization or bucketing, tokenization, part-of-speech tagging, SIFT feature extraction, removal of instances, processing of missing values)? If so, please provide a description. If not, you may skip the remaining questions in this section.

A1: Yes, the data underwent a filtering process to remove images with anomalous height distributions.

2. Was the "raw" data saved in addition to the preprocessed/cleaned/labeled data (e.g., to support unanticipated future uses)? If so, please provide a link or other access point to the "raw" data.

A2: No.

3. Is the software that was used to preprocess/clean/label the data available? If so, please provide a link or other access point.

A3: No.

B.5 Uses

1. Has the dataset been used for any tasks already? If so, please provide a description.

A1: No.

2. Is there a repository that links to any or all papers or systems that use the dataset? If so, please provide a link or other access point.

A2: N/A.

3. What (other) tasks could the dataset be used for?

A3: It can also be used for 3D change detection and disaster mapping applications.

4. Is there anything about the composition of the dataset or the way it was collected and preprocessed/cleaned/labeled that might impact future uses? For example, is there anything that a dataset consumer might need to know to avoid uses that could result in unfair treatment of individuals or groups (e.g., stereotyping, quality of service issues) or other risks or harms (e.g., legal risks, financial harms)? If so, please provide a description. Is there anything a dataset consumer could do to mitigate these risks or harms?

A4: No.

5. Are there tasks for which the dataset should not be used? If so, please provide a description.

A5: No.

B.6 Distribution

1. Will the dataset be distributed to third parties outside of the entity (e.g., company, institution, organization) on behalf of which the dataset was created? If so, please provide a description.

A1: Yes, SynRS3D and related codes will be made publicly available.

2. How will the dataset be distributed (e.g., tarball on website, API, GitHub)? Does the dataset have a digital object identifier (DOI)?

A2: We provide Zenodo download link for SynRS3D.

3. When will the dataset be distributed?

A3: Now we provide Zenodo download link for SynRS3D.

4. Will the dataset be distributed under a copyright or other intellectual property (IP) license, and/or under applicable terms of use (ToU)? If so, please describe this license and/or ToU, and provide a link or other access point to, or otherwise reproduce, any relevant licensing terms or ToU, as well as any fees associated with these restrictions.

A4: It will be distributed under the Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0) License.

5. Have any third parties imposed IP-based or other restrictions on the data associated with the instances? If so, please describe these restrictions, and provide a link or other access point to, or otherwise reproduce, any relevant licensing terms, as well as any fees associated with these restrictions.

A5: No.

6. Do any export controls or other regulatory restrictions apply to the dataset or to individual instances? If so, please describe these restrictions, and provide a link or other access point to, or otherwise reproduce, any supporting documentation.

A6: No.

B.7 Maintenance

1. Who will be supporting/hosting/maintaining the dataset?

A1: The authors.

2. How can the owner/curator/manager of the dataset be contacted (e.g., email address)?

A2: They can be contacted via email available on the GitHub repository.

3. Is there an erratum? If so, please provide a link or other access point.

A3: No.

4. Will the dataset be updated (e.g., to correct labeling errors, add new instances, delete instances)? If so, please describe how often, by whom, and how updates will be communicated to dataset consumers (e.g., mailing list, GitHub)?

A4: Yes, the authors will periodically review issues on GitHub and update the dataset based on the feedback.

5. If the dataset relates to people, are there applicable limits on the retention of the data associated with the instances (e.g., were the individuals in question told that their data would be retained for a fixed period of time and then deleted)? If so, please describe these limits and explain how they will be enforced.

A5: Not applicable.

6. Will older versions of the dataset continue to be supported/hosted/maintained? If so, please describe how. If not, please describe how its obsolescence will be communicated to dataset consumers.

A6: N/A.

7. If others want to extend/augment/build on/contribute to the dataset, is there a mechanism for them to do so? If so, please provide a description. Will these contributions be validated/verified? If so, please describe how. If not, why not? Is there a process for communicating/distributing these contributions to dataset consumers? If so, please provide a description.

A7: We have described the dataset generation process and all the tools used in detail in Section A.1. However, due to redistribution restrictions of the commercial add-ons used, we cannot provide the source code for synthetic data generation system. We are happy to assist anyone who wants to create or extend the dataset. Please contact the authors via email on our GitHub repository.

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[2] Gerald Baier, Antonin Deschemps, Michael Schmitt, and Naoto Yokoya. Synthesizing optical and sar imagery from land cover maps and auxiliary raster data. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–12, 2021.

[3] Wele Gedara Chaminda Bandara and Vishal M Patel. A transformer-based siamese network for change detection. In *IGARSS 2022-2022 IEEE International Geoscience and Remote Sensing Symposium*, pages 207–210. IEEE, 2022.

[4] Hao Chen and Zhenwei Shi. A spatial-temporal attention-based method and a new dataset for remote sensing image change detection. *Remote Sensing*, 12(10):1662, 2020.

[5] Hongruixuan Chen, Jian Song, Chengxi Han, Junshi Xia, and Naoto Yokoya. Changemamba: Remote sensing change detection with spatiotemporal state space model. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–20, 2024.

[6] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017.

[7] Gordon Christie, Kevin Foster, Shea Hagstrom, Gregory D Hager, and Myron Z Brown. Single view geocentric pose in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1162–1171, 2021.

[8] NTT DATA Corporation and Inc. DigitalGlobe. Aw3d high-resolution dataset. End User License Agreement, 2018. Available from NTT DATA Corporation and DigitalGlobe, Inc.

[9] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. *Advances in neural information processing systems*, 27, 2014.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[11] Shunping Ji, Shiqing Wei, and Meng Lu. Fully convolutional networks for multisource building extraction from an open aerial and satellite imagery data set. *IEEE Transactions on Geoscience and Remote Sensing*, 57(1):574–586, 2018.

[12] Bertrand Le Saux, Naoto Yokoya, Ronny Hänsch, and Myron Brown. 2019 ieee grss data fusion contest: large-scale semantic 3d reconstruction. *IEEE Geoscience and Remote Sensing Magazine (GRSM)*, 7(4):33–36, 2019.

[13] Yi Liu, Chao Pang, Zongqian Zhan, Xiaomeng Zhang, and Xue Yang. Building change detection for remote sensing images using a dual-task constrained deep siamese convolutional network model. *IEEE Geoscience and Remote Sensing Letters*, 18(5):811–815, 2020.

[14] Viktor Olsson, Wilhelm Tranheden, Juliano Pinto, and Lennart Svensson. Classmix: Segmentation-based data augmentation for semi-supervised learning. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 1369–1378, 2021.

[15] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

[16] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 12179–12188, 2021.

[17] Mario Fuentes Reyes, Yuxing Xie, Xiangtian Yuan, Pablo d'Angelo, Franz Kurz, Daniele Cerra, and Jiaojiao Tian. A 2d/3d multimodal data simulation approach with applications on urban semantic segmentation, building extraction and change detection. *ISPRS Journal of Photogrammetry and Remote Sensing*, 205:74–97, 2023.

[18] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[19] Franz Rottensteiner, Gunho Sohn, Jaewook Jung, Markus Gerke, Caroline Baillard, Sebastien Benitez, and Uwe Breitkopf. The isprs benchmark on urban object classification and 3d building reconstruction. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences; I-3*, 1(1):293–298, 2012.

[20] Jian Song, Hongruixuan Chen, and Naoto Yokoya. Syntheworld: A large-scale synthetic dataset for land cover mapping and building change detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 8287–8296, 2024.

[21] World Bank. World development indicators 2024, 2024.

[22] Junshi Xia, Naoto Yokoya, Bruno Adriano, and Clifford Broni-Bediako. Openearthmap: A benchmark dataset for global high-resolution land cover mapping. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 6254–6264, 2023.

[23] Yonghao Xu, Bo Du, Liangpei Zhang, Daniele Cerra, Miguel Pato, Emiliano Carmona, Saurabh Prasad, Naoto Yokoya, Ronny Hänsch, and Bertrand Le Saux. Advanced multi-sensor optical remote sensing for urban land use and land cover classification: Outcome of the 2018 ieee grss data fusion contest. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(6):1709–1724, 2019.

[24] Kunping Yang, Gui-Song Xia, Zicheng Liu, Bo Du, Wen Yang, Marcello Pelillo, and Liangpei Zhang. Semantic change detection with asymmetric siamese networks. *arXiv preprint arXiv:2010.05687*, 2020.

[25] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6023–6032, 2019.