# Georeferenced Cross-View 3D Reconstruction from Ground and Satellite Images

**David G. Shatwell** david.shatwell@ucf.edu
**Mubarak Shah** shah@crcv.ucf.edu

Institute of Artificial Intelligence, University of Central Florida

## Abstract

Georeferenced 3D reconstruction from ground images requires recovering not only the local scene geometry, but also localizing every camera center and 3D point in a geographic coordinate frame. Existing cross-view geo-localization methods are able to estimate the GPS coordinate of a single ground camera, but do not reconstruct dense 3D geometry, while recent feed-forward multi-view 3D reconstruction models predict point maps and camera poses only in a local coordinate frame. We introduce X-VGGT, a feed-forward cross-view geometry transformer that unifies these tasks by jointly processing a set of ground-level images and a georeferenced satellite image to predict ground-view camera poses, dense 3D point maps, and a scene-level transform parameterized by gravity, scale, yaw and translation. This transform projects the local reconstruction into the satellite plane, resulting in the geo-registration of all ground images with the georeferenced satellite image. We further propose an evaluation protocol for this joint task and show that, across multiple cross-view datasets, X-VGGT produces accurate georeferenced reconstructions and outperforms prior cross-view geo-localization and multi-view 3D reconstruction models. Project page: `https://dshatwell23.github.io/xvggt-page/`.

## 1 Introduction

Ground-level images are easy to collect and can be used to reconstruct detailed 3D models; however, image-based reconstruction methods typically recover geometry only up to an unknown global scale. In recent years, high-resolution satellite images have become widely available across many regions of the world. Since these images are often georeferenced, their image coordinates can be mapped to real-world geographic locations, providing GPS information for each pixel. This makes them an ideal complementary source of information for image-based 3D reconstruction, as they can provide not only a metric reference for resolving scale ambiguity, but also a global coordinate frame for localizing the reconstructed cameras and 3D points.

However, using satellite images as a geographic reference requires solving a difficult cross-view registration problem: ground-level and satellite images observe the same scene from drastically different viewpoints, with large changes in appearance, scale, and visible structure. A model that can overcome this gap would enable practical applications in autonomous driving and robotics in GPS-denied areas, disaster response from localized post-event imagery, and infrastructure inspection, where observed assets and defects could be placed directly on a map.

Prior work addresses important parts of this problem, but not the full setting. Cross-view geo-localization methods are trained to estimate ground-view camera locations within a satellite image [35, 50, 37, 36, 54, 55, 20], but they are not designed to localize individual image pixels. This capability is known as geo-registration, and requires additional 3D understanding about the scene geometry. Recent feed-forward 3D reconstruction models are able to recover camera poses and 3D

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

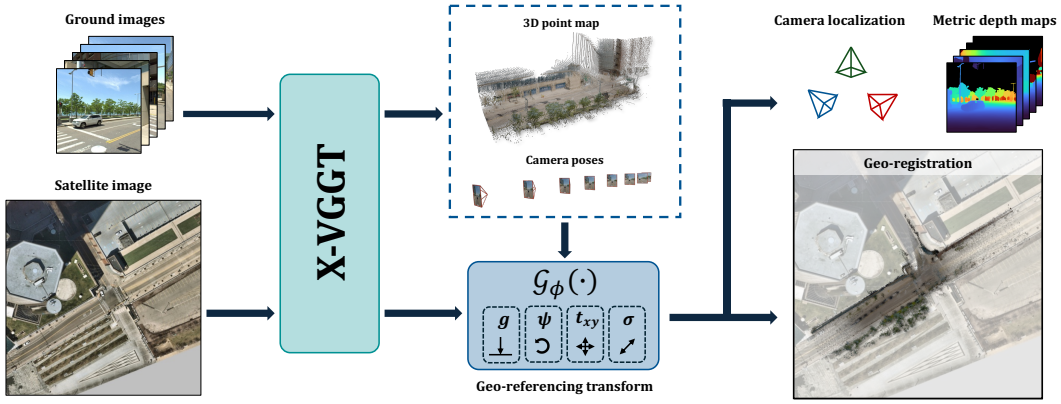


Figure 1: **Georeferenced cross-view 3D reconstruction.** Given a set of ground-level images and a georeferenced satellite view, **X-VGGT** predicts camera poses, 3D point-maps, and a georeferencing transform $\mathcal{G}_\phi(\cdot)$. By projecting the 3D reconstruction into the satellite plane, our model can geo-register every pixel in the ground images, localize their cameras, and generate metric depth maps.

point maps from sparse views [49, 21, 46, 24, 51], but they typically generate predictions in a local coordinate frame and are known to fail under extreme viewpoint changes.

Georeferenced cross-view 3D reconstruction lies at the intersection of these two tasks: using both ground and satellite images, the goal is to generate a metric 3D reconstruction where all ground-level image pixels are localized in a geographic coordinate frame. To address this problem, we introduce **X-VGGT**, a Cross(**X**)-**V**iew **G**eometry **G**rounded **T**ransformer for georeferenced cross-view 3D reconstruction. Given a set of ground images and a georeferenced satellite image, X-VGGT predicts ground-view camera extrinsics, intrinsics, metric dense 3D point maps, and a scene-level satellite alignment transform used to geo-localize image pixels and camera centers, as shown in Figure 1. The satellite image is processed jointly with the ground images through a multi-view transformer, allowing them to exchange information. Then, a dedicated Satellite Alignment Head predicts a ground-to-satellite coordinate transform parameterized by gravity, yaw, scale, and translation, which projects the reconstructed camera centers and dense point maps into the georeferenced satellite plane, thus enabling dense pixel-level geo-registration.

In summary, our key contributions are as follows:

- We propose X-VGGT, a feed-forward cross-view 3D reconstruction model that jointly uses ground images and a georeferenced satellite image to predict cameras, metric 3D point maps, and a satellite-ground alignment transform that geo-localizes all ground image pixels.
- We enable georeferenced 3D reconstruction from flexible sets of ground-view inputs: X-VGGT grounds camera centers and dense point maps in the satellite coordinate frame while supporting ordered sequences or unordered sets of limited-FOV images.
- We introduce a Satellite Alignment Head and georeferencing losses that directly supervise the ground-to-satellite transformation, projected camera centers, and pixel-wise geo-registration.
- We introduce new training and evaluation protocols for cross-view 3D reconstruction, using ordered and unordered sets of frames from unseen locations to evaluate the generalization of X-VGGT and competing baselines.

## 2 Related Work

**Geo-localization** aims to estimate where an image was captured by predicting the camera GPS coordinates using only visual information. Existing approaches are broadly grouped into classification, retrieval, or registration. Classification methods discretize the Earth into geographic cells and use the centroid of the predicted geocell as the estimated GPS coordinate [53, 42, 32, 27, 30, 10]. Retrieval methods embed the query into a high-dimensional feature space and compare it against a geotagged gallery, formed either by encoding GPS coordinates with a location encoder [41, 19, 33, 5, 15, 4, 34], or by using images with known locations as proxies.

Image-retrieval methods can be divided into same-view and cross-view settings. Same-view visual place recognition (VPR) embeds query and gallery images with a shared image encoder and ranks the gallery by feature similarity [39, 3, 6, 1, 2, 17, 7, 40]. Cross-view geo-localization instead matches ground-level queries against satellite images, using either shared or view-specific encoders. Prior work has improved this framework with ground-view warping, hard negative mining, re-ranking, stronger backbones, and auxiliary objectives [63, 11, 22, 62, 58]. Cross-view retrieval has also been extended to videos using multi-stage pipelines and temporal aggregation [61, 44, 29]; however, these methods typically assume ordered image sequences and are less suited to unstructured multi-view scenes.

A key limitation of retrieval-based geo-localization is that predictions remain tied to the gallery discretization: the estimated GPS coordinates represent the center pixel of a query, rather than pixel-wise GPS locations of the query image. Recent cross-view localization methods address this by registering the ground view within a georeferenced satellite image, typically through pose search, BEV transformations, dense correspondence fields, or fine-grained cross-view matching [35, 54, 50, 37, 47, 20, 36, 55, 56]. However, they remain primarily 2D pose-estimation systems: most predict a single rotation and translation within a fixed satellite area from a single ground image or panorama, without reconstructing dense 3D structure or placing multiple cameras and point maps in a shared georeferenced frame. This motivates moving beyond local 2D registration toward georeferenced multi-view 3D reconstruction.

**Multi-view 3D reconstruction** recovers camera geometry and dense scene structure rather than only a 2D pose. Recent feed-forward geometry models directly predict this structure from images: DUSt3R [49] and MASt3R [21] estimate point maps and correspondences from uncalibrated image pairs, and several methods extend this paradigm to larger or more structured inputs through multi-view prediction, memory, recurrent state, relative-pose aggregation, or video geometry [9, 48, 12, 59]. In parallel, learning-based SfM and general multi-view reconstruction models are increasingly adapted to predict cameras, depth, point maps, visual appearance, or metric 3D from sparse or dense scenes [45, 46, 24, 51, 57, 60, 18].

However, these models are not designed for georeferenced ground-satellite reconstruction. They typically recover geometry in a local coordinate frame with arbitrary scale and rely on sufficient visual overlap, whereas nadir satellite and street-level images differ substantially in viewpoint, appearance, and visibility. Recent methods partially address this gap by adapting 3D reconstruction models to drone-ground pairs [43], drone-satellite pairs [25], ground images and top-down floor plans [16], or satellite, aerial, and ground views [52]. While promising, these approaches do not directly solve ground-satellite registration with real orthorectified satellite imagery: they either rely on drone viewpoints, use non-standard top-down inputs, or are trained with synthetic perspective-projection aerial images.

While extreme-view methods make important progress, they do not fully address the setting targeted by our method: reconstructing arbitrary ordered or unordered sets of ground views directly in the coordinate frame of a georeferenced orthorectified satellite image, without using any auxiliary modalities. In this work, our aim is to connect cross-view localization with multi-view 3D reconstruction, using satellite images to geo-register all cameras and pixels in a geographic reference frame.

# 3 Method

Given a set of ground images and a georeferenced satellite image, X-VGGT predicts ground-view camera poses, dense 3D point maps, and a satellite-to-ground transform that is used to geo-register all cameras and image pixels. In this section, we first formulate the georeferenced 3D reconstruction problem (Sec. 3.1), then describe the architecture (Sec. 3.2), training objectives (Sec. 3.4), and training strategy (Sec. 3.5). Figure 2 provides an overview of our method.

## 3.1 Problem Formulation

Let $I_{\mathrm{sat}}$ denote an orthorectified georeferenced satellite image, where image coordinates can be mapped to latitude-longitude pairs, and let $\{I_i\}_{i=1}^{N}$ denote $N$ ground-level images captured within the satellite area but with unknown geo-location. The ground inputs may be an ordered sequence, an unordered set, or perspective views extracted from a panorama. X-VGGT predicts camera extrinsics

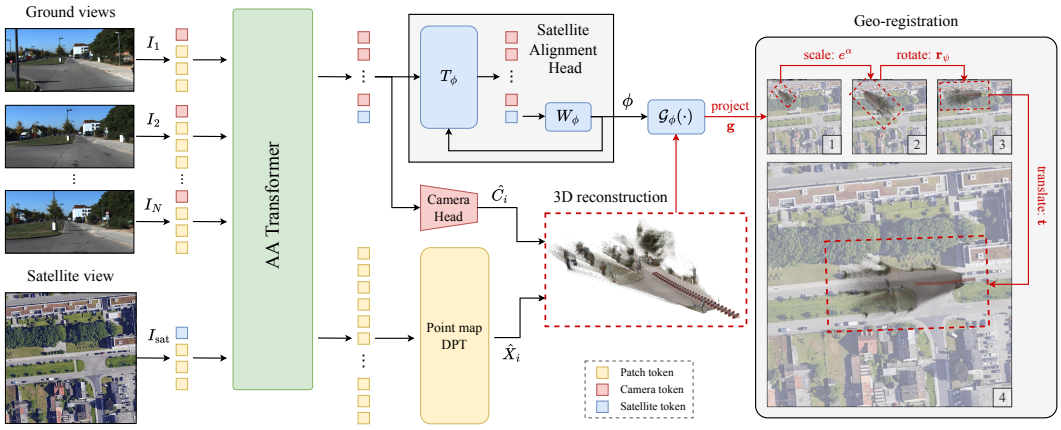


Figure 2: **Overview of X-VGGT.** The model jointly processes $N$ ground-level images and a georeferenced satellite image with an Alternating Attention (AA) Transformer. Ground-view patch tokens are decoded by the camera and DPT heads to predict camera poses $\hat{C}_i$ and dense point maps $\hat{X}_i$, while the satellite token is decoded by the Satellite Alignment Head (SAH) to estimate the transformation $\mathcal{G}_\phi$ from the local 3D reconstruction to the satellite coordinate frame. $\mathcal{G}_\phi$ is parameterized by a gravity vector $\mathbf{g}$ for ground-plane projection, a scalar $\alpha$ for metric conversion, and a planar rotation $\psi$ and translation $\mathbf{t}$ that align the projected points with the satellite image.

$\hat{E}_i \in \mathbb{R}^{3\times4}$, intrinsics $\hat{K}_i \in \mathbb{R}^{3\times3}$, dense 3D point maps $\hat{X}_i \in \mathbb{R}^{H\times W\times3}$, and a georeferencing transform $\mathcal{G}_\phi : \mathbb{R}^3 \to [0,1]^2$ used to warp pixels from the local ground-views into the satellite image plane, allowing dense point maps and camera centers to be projected and geo-localized.

The georeferencing map is a single scene-level transform. Rather than predicting each camera location independently, X-VGGT predicts one geometric map that aligns the full local reconstruction with the satellite plane. We parameterize this map by

$$\phi = (\mathbf{g}, \mathbf{r}_\psi, \alpha, \mathbf{t}), \qquad \mathbf{r}_\psi = (\cos\psi, \sin\psi),$$

where $\mathbf{g} \in \mathbb{R}^3$ is a unit gravity vector in the local reconstruction frame, $\mathbf{r}_\psi \in \mathbb{R}^2$ is a unit yaw vector that rotates the horizontal basis, $\alpha = \log s$ is the logarithmic scaling factor, and $\mathbf{t} \in \mathbb{R}^2$ is a translation vector in normalized satellite coordinates.

The gravity direction defines the normal of the satellite-aligned plane. To obtain in-plane axes, we project a canonical reference direction onto the plane orthogonal to $\mathbf{g}$. Let $\mathbf{a}$ be the normalized projection of the $x$-axis onto this plane and let $\mathbf{b} = \mathbf{g} \times \mathbf{a}$; the yaw vector is used to rotate the basis as follows:

$$\mathbf{x}_\phi = \cos\psi\,\mathbf{a} + \sin\psi\,\mathbf{b}, \qquad \mathbf{y}_\phi = -\sin\psi\,\mathbf{a} + \cos\psi\,\mathbf{b}.$$

For any reconstructed point $\mathbf{X}$, the satellite projection is thus given by

$$\mathcal{G}_\phi(\mathbf{X}) = \exp(\alpha) \begin{bmatrix} \mathbf{x}_\phi^\top \mathbf{X} \\ \mathbf{y}_\phi^\top \mathbf{X} \end{bmatrix} + \mathbf{t}.$$

This orthographic projection maps camera centers and dense points from the local reconstruction frame into normalized satellite coordinates, with the vertical axis pointing upward. For a satellite crop spanning $\rho_{\text{sat}}$ meters, $\rho_{\text{sat}} e^\alpha$ gives the metric scale in meters per reconstruction unit.

### 3.2 Cross-View Geometry Grounded Transformer

X-VGGT jointly processes the ground and satellite images with a multi-view transformer using alternating attention layers. The satellite image is appended as the final view, giving the input tuple $(I_1, \ldots, I_N, I_{\text{sat}})$ with $M = N + 1$ total views. A shared visual encoder, initialized from DINOv2 [28], converts each image into $P$ patch tokens ($\mathbf{p}_i$). Camera ($\mathbf{c}_i$) and register ($\mathbf{z}_{i,j}^{\text{reg}}$) tokens are appended to every view. All tokens have embedding dimension $D = 1024$:

$$Z_i^0 = [\,\mathbf{c}_i, \mathbf{z}_{i,1}^{\text{reg}}, \ldots, \mathbf{z}_{i,N_r}^{\text{reg}}, \mathbf{p}_{i,1}, \ldots, \mathbf{p}_{i,P}\,] \in \mathbb{R}^{(1+N_r+P)\times D}, \qquad i = 1, \ldots, M.$$

We use three sets of learnable camera and register tokens: one for the reference ground view $I_1$, one shared by the remaining ground views $\{I_i\}_2^N$, and one for the satellite view $I_{\text{sat}}$. A shared image encoder produces the patch tokens for all views.

Following the alternating-attention design of VGGT [46], each block alternates between frame attention, which updates tokens within each view, and global attention, which updates tokens across the concatenated set of views:

$$\bar{Z}_i^\ell = F_\ell(Z_i^{\ell-1}), \qquad Z^\ell = G_\ell([\,\bar{Z}_1^\ell; \ldots; \bar{Z}_M^\ell\,]).$$

This preserves view-specific image evidence while repeatedly exchanging information across the ground and satellite views.

The satellite image has a different geometric role from a ground camera. Since the satellite image is orthorectified and used as a nadir map, X-VGGT does not predict a perspective camera pose, finite camera center, depth map, or point map. Instead, the satellite tokens condition the ground 3D reconstruction through attention and are read by a dedicated Satellite Alignment Head.

**Satellite Alignment Head.** The Satellite Alignment Head (SAH) predicts scene-level georeferencing parameters $\phi$. Let $Y \in \mathbb{R}^{M\times 2D}$ denote the camera-token features from all $M$ views, including satellite. Each row concatenates the $D$-dimensional camera-token embeddings from the final frame-attention and global-attention blocks, yielding $2D = 2048$ features per view. At each refinement step $k$, the head embeds the previous estimate $\tilde{\phi}^{k-1}$, mixes the $M$ camera tokens with a lightweight transformer $T_\phi$, and reads out only the final satellite token:

$$U^k = T_\phi(Y; \tilde{\phi}^{k-1}), \qquad \tilde{\phi}^k = \tilde{\phi}^{k-1} + W_\phi(U^k_{\text{sat}}), \qquad k = 1, \ldots, 4.$$

For the first step, $\tilde{\phi}^0$ is represented by a learned empty georeferencing token. The final activated estimate $\phi = A(\tilde{\phi}^4)$ normalizes $\mathbf{g}$ and $\mathbf{r}_\psi$ to unit length while leaving $\alpha$ and $\mathbf{t}$ unconstrained. Reading only from the satellite token reflects the role of $\phi$: it describes the alignment between the reconstructed ground scene and the satellite frame.

**3D Geometry Heads.** The ground-view geometry heads follow the VGGT design. The camera head uses an iterative transformer over the $N$ ground camera tokens and predicts a pose encoding in $\mathbb{R}^{N\times 9}$, consisting of 3D translation, a 4D quaternion rotation, and 2D field-of-view parameters for each ground view. The point head is a dense prediction transformer (DPT) [31] that converts ground patch tokens into dense point maps and confidence maps, $\hat{X} \in \mathbb{R}^{N\times H\times W\times 3}$. Both of these heads receive only the ground-view tokens as input; the satellite patch tokens are not directly ingested by any downstream head, and instead are used as context during the joint encoding stage.

### 3.3 Geo-registration

The camera head predicts a world-to-camera transform $\hat{E}_i = [\hat{R}_i \mid \hat{\mathbf{t}}_i]$ for each ground view in the local reconstruction frame. We recover the camera center as $\hat{\mathbf{C}}_i = -\hat{R}_i^\top \hat{\mathbf{t}}_i$ and project both dense point maps and camera centers into the satellite image:

$$\hat{\mathbf{u}}_i(p) = \mathcal{G}_\phi(\hat{\mathbf{X}}_i(p)), \qquad \hat{\mathbf{u}}_i^{\text{cam}} = \mathcal{G}_\phi(\hat{\mathbf{C}}_i),$$

where $p$ indexes image pixels. The projected camera centers provide a fine-grained localization signal, while the projected dense points measure whether the reconstructed geometry is correctly grounded in the satellite frame.

### 3.4 Training Objectives

X-VGGT is trained with a multi-task objective that combines standard 3D reconstruction losses with new georeferencing losses. The georeferencing losses supervise the scene-level map parameters and their induced satellite projections. Let $\phi^*$ denote the ground-truth georeferencing parameters computed in the normalized reconstruction frame. For each Satellite Alignment Head (SAH) refinement stage $k$, we supervise the activated estimate $\phi^k = (\mathbf{g}^k, \mathbf{r}_\psi^k, \alpha^k, \mathbf{t}^k)$ with

$$\mathcal{L}_\phi = \frac{1}{K_\phi} \sum_{k=1}^{K_\phi} \gamma^{K_\phi - k} \left[1 - \langle \mathbf{g}^k, \mathbf{g}^* \rangle + 1 - \langle \mathbf{r}_\psi^k, \mathbf{r}_\psi^* \rangle + |\alpha^k - \alpha^*| + \|\mathbf{t}^k - \mathbf{t}^*\|_1\right],$$

where $K_\phi = 4$ and $\gamma = 0.6$. The direction terms supervise the unit gravity and yaw vectors, while the remaining terms supervise scale and translation.

We also supervise the satellite projections induced by the final georeferencing transform. Specifically, we compare both the dense projected point maps and the projected camera centers against their ground-truth normalized satellite coordinates. The dense loss is computed over valid pixels $\Omega_{\text{pts}}$, while the camera-center loss is computed over the $N$ ground-view cameras:

$$\mathcal{L}_{\text{dense}} = \frac{1}{|\Omega_{\text{pts}}|} \sum_{(i,p)\in\Omega_{\text{pts}}} \left\| \mathcal{G}_\phi(\hat{\mathbf{X}}_i(p)) - \mathbf{u}_i(p) \right\|_1, \quad \mathcal{L}_{\text{center}} = \frac{1}{N} \sum_{i=1}^{N} \left\| \mathcal{G}_\phi(\hat{\mathbf{C}}_i) - \mathbf{u}_i^{\text{cam}} \right\|_1.$$

The remaining losses follow standard feed-forward 3D reconstruction training. The camera loss $\mathcal{L}_{\text{cam}}$ applies an L1 loss to the predicted pose encoding, decomposed into translation, quaternion rotation, and field-of-view terms. The point-map loss $\mathcal{L}_{\text{pts}}$ combines confidence-weighted point regression and a normal consistency term over valid pixels. The final objective is given by

$$\mathcal{L} = \lambda_{\text{cam}}\mathcal{L}_{\text{cam}} + \lambda_{\text{pts}}\mathcal{L}_{\text{pts}} + \lambda_\phi\mathcal{L}_\phi + \lambda_{\text{center}}\mathcal{L}_{\text{center}} + \lambda_{\text{dense}}\mathcal{L}_{\text{dense}}.$$

All georeferencing and point-map terms have unit weight except for the camera loss, which uses a weight of $5.0$.

### 3.5 Training Strategy

**Supervision signals.** Each training sample contains calibrated ground images, depth-derived point maps, ground camera poses in an East-North-Up (ENU) reference frame, and an orthorectified satellite image with known spatial resolution. During training, the data-loaders project the ENU camera centers into the satellite image to obtain normalized camera-center targets $\mathbf{u}_i^{\text{cam}} \in [0,1]^2$. After the standard normalization of camera poses and point maps into the local reconstruction frame, we recompute all georeferencing targets in the same coordinate system as the model predictions. Let $E_i = [R_i \mid \mathbf{t}_i]$ be the normalized world-to-camera pose, $\mathbf{C}_i = -R_i^\top \mathbf{t}_i$ the camera center, and $\mathbf{e}_z = (0,0,1)^\top$ the ENU up direction. If $R_0^{\text{enu}}$ is the reference-camera rotation before normalization and $\bar{R}_0$ is the reference-camera rotation after normalization, the ground-truth gravity in the normalized reconstruction frame is $\mathbf{g}^* = \text{norm}(\bar{R}_0^\top R_0^{\text{enu}} \mathbf{e}_z)$.

Given $\mathbf{g}^*$, we solve for the yaw, scale, and translation that best align the normalized ground camera centers with their satellite coordinates:

$$(\mathbf{r}_\psi^*, \alpha^*, \mathbf{t}^*) = \arg\min_{\mathbf{r}_\psi,\alpha,\mathbf{t}} \sum_{i=1}^{N} \left\| \mathcal{G}_{(\mathbf{g}^*,\mathbf{r}_\psi,\alpha,\mathbf{t})}(\mathbf{C}_i) - \mathbf{u}_i^{\text{cam}} \right\|_2^2,$$

Dense satellite targets are then generated by projecting each valid depth-derived 3D point through the fitted transform $\mathbf{u}_i(p) = \mathcal{G}_{\phi^*}(\mathbf{X}_i(p))$, for $p \in \Omega_{\text{pts}}$.

**Training datasets.** We train our model with KITTI [14, 35] and a modified version of Aerial-MegaDepth [43], which we call Sat-MegaDepth. KITTI provides street-level sequences with calibrated camera poses, LiDAR-derived depth maps, and georeferenced satellite views. We construct Sat-MegaDepth from Aerial-MegaDepth [43] by adding orthorectified satellite images, which are not provided by the original dataset. Unlike KITTI, MegaDepth-style images are scattered across a scene rather than arranged as a vehicle trajectory. Training with both datasets encourages X-VGGT to handle sequential or unordered ground-view sets. Additional details on training data, satellite crops, augmentations, dataset construction, and optimization are provided in Appendix C.

## 4 Experiments

We evaluate X-VGGT on three georeferenced reconstruction tasks: camera geo-localization, pixel-wise geo-registration, and local 3D reconstruction. We first describe the datasets, scene construction, baselines, and metrics used throughout the section, then report quantitative results and ablations.

### 4.1 Evaluation Datasets

We evaluate X-VGGT on four cross-view datasets spanning different locations, sensors, and scene layouts. KITTI [14, 35] provides driving sequences with LiDAR-derived depth and georeferenced

satellite images; although KITTI is used during training, evaluation is performed on the geographically disjoint cross-area Test-2 split. We use the augmented KITTI release of Shi and Li [35], which supplements the original dataset with Google Maps satellite imagery at zoom level 18, corresponding to a spatial resolution of 0.20 m/pixel. Mapillary Metropolis provides synchronized multi-directional driving images, camera poses, MVS-derived depth, and satellite images, and is used for zero-shot evaluation. We further evaluate camera geo-localization on WRIVA [8] and ACC-NVS1 [38], which contain unseen locations with calibrated ground images and GPS annotations but no ground-truth depth maps. Additional dataset and satellite-image construction details are provided in Appendix D.

### 4.2 Scene Generation

We construct evaluation scenes according to the specific scene layout of each dataset: for KITTI and ACC-NVS1, we use sequences of consecutive video frames; for Mapillary Metropolis, we use consecutive multi-directional camera groups pointing along the front, back, left, and right sides of the vehicle frame; and for WRIVA, we create unordered image sets with similar camera poses. Each scene consists of $N$ ground images and a $150\mathrm{m} \times 150\mathrm{m}$ satellite view sampled around the selected ground cameras. We create satellite crops such that the centroid of the ground cameras is randomly placed within the central $100\mathrm{m} \times 100\mathrm{m}$ region of the satellite image. We sample up to $1024$ scenes per dataset for the main evaluation and use larger scenes with up to $96$ ground images for the ablations. Additional scene construction details are provided in Appendix D.

### 4.3 Baselines and Evaluation Protocols

We compare X-VGGT against cross-view geo-localization methods, multi-view 3D reconstruction models, and a baseline that aligns reconstructed geometry using predicted camera geo-locations. All methods are evaluated using the same scene inputs: 16 ground images and one georeferenced satellite image. Table 1 summarizes the native prediction capabilities of the individual models.

Table 1: Native prediction support for each method.

| Method | Cam. Geo. | Pixel Geo. | 3D Recon. |
|---|---|---|---|
| FG$^2$ [55] | ✓ | ✗ | ✗ |
| Loc$^2$ [56] | ✓ | ✓ | ✗ |
| MASt3R [21] | ✗ | ✗ | ✓ |
| VGGT [46] | ✗ | ✗ | ✓ |
| A-MD [43] | ✗ | ✗ | ✓ |
| **X-VGGT** | ✓ | ✓ | ✓ |

**Cross-view geo-localization.** FG$^2$ [55] and Loc$^2$ [56] estimate a ground camera's planar location and orientation within a satellite image. Each ground image is processed independently with the scene's satellite image. We report the north-up setting, without a ground-truth heading prior, and aggregate predictions over the ground views. For Loc$^2$, dense geo-registration is obtained by projecting its predicted depth using the estimated camera pose.

**3D reconstruction baselines.** We evaluate MASt3R [21], VGGT [46], and Aerial-MegaDepth [43] using both the ground and satellite images. The satellite is placed at index 0 as the reference view. VGGT jointly processes all 17 images, while MASt3R and Aerial-MegaDepth use a satellite-centered star graph pairing the satellite with each ground image, followed by sparse global alignment. Since these models only support perspective views, we predict a virtual perspective camera for the satellite reference, defined by it's intrinsic $\hat{K}_{\mathrm{sat}}$ and extrinsic $[\hat{R}_{\mathrm{sat}} \mid \hat{\mathbf{t}}_{\mathrm{sat}}]$ parameters. We obtain satellite coordinates for each reconstructed camera center or dense point $\mathbf{X}$ using

$$\hat{\mathbf{u}}(\mathbf{X}) = \mathcal{N}\Big[\pi\Big(\hat{K}_{\mathrm{sat}}\big(\hat{R}_{\mathrm{sat}}\mathbf{X} + \hat{\mathbf{t}}_{\mathrm{sat}}\big)\Big)\Big],$$

where $\pi$ denotes perspective projection and $\mathcal{N}$ converts image coordinates to normalized satellite coordinates. No ground-truth pose, scale, or Sim(3) alignment is applied when computing geographic predictions for these baselines.

**Geo-localization-aligned reconstruction.** Additionally, we evaluate a stronger baseline by aligning the 3D reconstruction and relative camera poses predicted by VGGT with the camera geo-locations independently predicted by Loc$^2$. VGGT reconstructs the 16 ground views without satellite, while Loc$^2$ independently predicts each ground camera's location in the satellite image. We estimate a common gravity direction from VGGT's predicted camera orientations and project the reconstruction into the ground plane. We then use RANSAC [13] to fit a planar similarity transform between the projected camera centers and Loc$^2$'s predicted geo-locations. This defines a gravity-constrained Sim(3) transform, with zero vertical translation, that is applied consistently to all reconstructed cameras and dense 3D points. The alignment's yaw is inferred from camera-center correspondences, without using Loc$^2$'s per-frame yaw predictions.

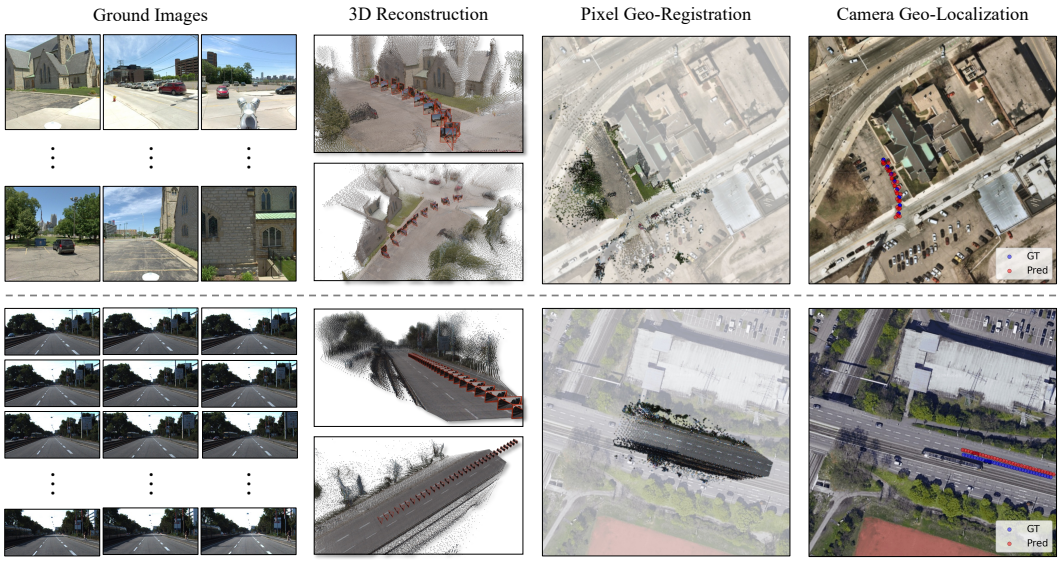


Figure 3: **Qualitative results.** We show two sample predictions from the Metropolis (top) and KITTI (bottom) datasets using X-VGGT. From left to right: input ground views, predicted 3D reconstruction, pixel-level geo-registration, and camera geo-localization. Our method jointly reconstructs the ground-view scene while projecting both image pixels and camera centers into the satellite coordinate frame.

**Evaluation policy.** We convert normalized satellite-coordinate errors to meters using the known satellite resolution. Predicted coordinates are not clipped to the satellite boundary, which means that incorrect predictions can produce errors exceeding the 150 m satellite extent.

### 4.4 Evaluation Metrics

**Camera geo-localization** measures whether the predicted camera centers and orientation are correctly grounded in the satellite image. Since X-VGGT predicts camera locations in normalized satellite coordinates $\hat{\mathbf{u}}_i, \mathbf{u}_i \in [0,1]^2$, we define the location error as $e_i^{\text{cam}} = \|(\hat{\mathbf{u}}_i - \mathbf{u}_i)\|_2 \, \rho_{\text{sat}}$, where $\rho_{\text{sat}} = 150$ is the satellite scale in meters. Orientation error is computed by projecting each camera's principal ray into the satellite plane and measuring the angle between the predicted and ground-truth 2D directions.

**Pixel geo-registration** measures the distance between the ground truth and predicted pixel geo-locations. We compute this metric by projecting the image pixels into 3D space, and then into the satellite plane using the ground truth georeferencing transform. We report the location error between predicted and ground-truth satellite coordinates over valid pixels whose ground-truth projections lie inside the satellite image.

**3D reconstruction** evaluates the local multi-view geometry independently of satellite alignment. We align predicted camera poses to the metric ground-truth cameras with a similarity transform, then compute pairwise relative pose errors. We report relative rotation accuracy (RRA), relative translation accuracy (RTA), and their joint AUC at thresholds up to $30^\circ$.

### 4.5 Experimental Results

**Camera Geo-Localization.** Table 2 compares camera geo-localization across four datasets. X-VGGT achieves the lowest camera-center and orientation errors among the individual localization and reconstruction baselines, with average errors of $22.38$ m and $30.93^\circ$. Cross-view localization methods remain competitive for predicting the camera-centers but produce large orientation errors, whereas 3D reconstruction baselines sometimes estimate orientation more accurately but have substantially larger localization errors. The largest localization gain occurs on Metropolis, where X-VGGT generalizes to an unseen multi-directional camera setup. Compared with geo-localization-aligned 3D reconstruction baseline (VGGT + Loc$^2$), our method reduces localization error by $6.84$ m on KITTI

Table 2: **Camera geo-localization.** Mean camera-center localization error in meters and orientation error in degrees across four cross-view datasets. KITTI is evaluated on a geographically disjoint test split, while Metropolis, WRIVA, and ACC-NVS1 are used for zero-shot evaluation. VGGT + Loc$^2$ denotes geo-localization-aligned reconstruction, which aligns VGGT's reconstructed camera centers to locations independently predicted by Loc$^2$ using a shared Sim(3) transform. Dashes indicate results not reported.

| Method | KITTI | | Metropolis | | WRIVA | | ACC-NVS1 | |
|---|---|---|---|---|---|---|---|---|
| | Loc. (m) | Orien. (°) | Loc. (m) | Orien. (°) | Loc. (m) | Orien. (°) | Loc. (m) | Orien. (°) |
| FG$^2$ [55] | 39.81 | 89.00 | 40.26 | 89.73 | 36.47 | 85.80 | 36.51 | 77.58 |
| Loc$^2$ [56] | 32.00 | 69.99 | 37.61 | 84.09 | 36.30 | 76.16 | 39.96 | 80.81 |
| MASt3R [21] | 1103.32 | 62.22 | 1028.05 | 79.83 | 1446.36 | 88.89 | 952.52 | 86.13 |
| VGGT [46] | 130.05 | 32.17 | 408.13 | 60.77 | 455.36 | 63.00 | 168.45 | 77.54 |
| A-MD [43] | 104.95 | 24.68 | 154.35 | 75.82 | 412.00 | 79.24 | 190.72 | 74.84 |
| VGGT + Loc$^2$ | 31.46 | 81.16 | 37.06 | 89.19 | 35.16 | 87.41 | 37.07 | 89.41 |
| **X-VGGT (Ours)** | **24.62** | **22.50** | **14.35** | **17.59** | **26.05** | **40.64** | **24.49** | **42.99** |

Table 3: **Pixel-wise geo-registration on datasets with depth.** Mean geo-localization error (meters) over ground-image pixels for KITTI and Mapillary Metropolis.

| Method | KITTI | Metropolis |
|---|---|---|
| Loc$^2$ [56] | 43.96 | 88.96 |
| MASt3R [21] | 900.85 | 1500.67 |
| VGGT [46] | 146.35 | 640.16 |
| A-MD [43] | 62.07 | 319.15 |
| VGGT + Loc$^2$ | 178.45 | 50.54 |
| **X-VGGT (Ours)** | **25.52** | **17.84** |

Table 4: **Local 3D reconstruction.** AUC of joint RRA and RTA up to 30°. X-VGGT preserves the strong local reconstruction behavior of VGGT while adding georeferencing capabilities.

| Method | KITTI | Metropolis | WRIVA | ACC-NVS1 |
|---|---|---|---|---|
| MASt3R [21] | 6.64 | 5.05 | 0.03 | 6.25 |
| VGGT [46] | 83.29 | **55.31** | 16.90 | 67.96 |
| A-MD [43] | 21.58 | 9.20 | 4.13 | 8.55 |
| **X-VGGT (Ours)** | **83.90** | 51.34 | **21.17** | **72.91** |

and 22.71 m on Metropolis, while reducing orientation error from 81.16° to 22.50° and from 89.19° to 17.59°, respectively. Overall, these results show that neither 2D cross-view localization, local 3D reconstruction or their combination are sufficient; robust georeferenced reconstruction requires jointly estimating scene geometry and its alignment to the satellite frame.

**Pixel-Wise Geo-Registration.** Table 3 evaluates dense pixel geo-registration on KITTI and Metropolis, the two datasets with depth annotations. X-VGGT achieves the lowest errors, improving over the strongest baseline by 18.44 m and 32.70 m, respectively. The closest competitors are Loc$^2$ on KITTI and VGGT + Loc$^2$ on Metropolis. Compared directly with VGGT + Loc$^2$, X-VGGT reduces error from 178.45 m to 25.52 m on KITTI and from 50.54 m to 17.84 m on Metropolis. Although geo-localization-aligned reconstruction improves upon standalone Loc$^2$ on Metropolis, it substantially worsens dense registration on KITTI despite slightly improving camera-center localization. This indicates that fitting reconstructed camera centers to predicted geographic locations does not by itself ensure accurate geo-registration of the dense point maps, supporting joint estimation of scene geometry and satellite alignment.

**Local 3D Reconstruction.** Table 4 shows that X-VGGT remains competitive on local 3D reconstruction, despite being trained to also estimate satellite alignment. It slightly improves over VGGT in most settings, likely because our model is initialized from the VGGT checkpoint while being further adapted to ground-level multi-view scenes. This indicates that adding georeferencing supervision does not degrade local 3D reconstruction quality.

**Qualitative Results.** Figure 3 provides qualitative examples of X-VGGT predictions. The results show that our method produces coherent local 3D reconstructions while accurately projecting both dense points and camera centers into the satellite frame.

Table 5: **Effect of georeferencing losses on KITTI [14].** Adding dense point supervision and camera supervision improves both camera-level and pixel-level geolocation.

| Losses | Cam. Pose | | 3D Points |
|---|---|---|---|
| | Loc. (m) | Orien. (°) | Loc. (m) |
| $\mathcal{L}_\phi$ | 30.35 | 32.87 | 33.17 |
| + $\mathcal{L}_{\text{dense}}$ | 27.95 | 34.49 | 30.91 |
| + $\mathcal{L}_{\text{center}}$ | **27.24** | **30.73** | **29.45** |

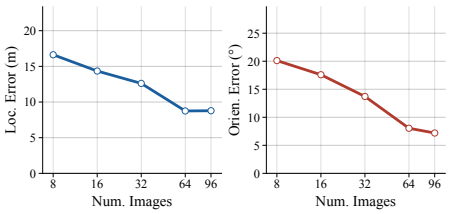


Figure 4: **Effect of the number of ground frames on Metropolis.** Additional ground views reduce both camera localization and orientation errors.

**Ablations on the Effect of Georeferencing Losses.** Table 5 ablates the proposed georeferencing losses on the KITTI dataset. Supervising only the georeferencing parameters provides a reasonable initial alignment, but adding dense projection supervision reduces the pixel-wise geo-registration error and improves camera localization. Adding camera-center supervision further improves the alignment, yielding the lowest errors across all reported metrics. These results show that the parameter, dense, and camera-center losses provide complementary signals for reducing both camera geo-localization and pixel-wise geo-registration errors.

**Ablations on the Effect of the Number of Ground Images.** Figure 4 evaluates how performance changes with the number of input ground views on Mapillary Metropolis. Both localization and orientation errors decrease as more frames are provided, showing that X-VGGT effectively uses additional multi-view context to estimate a more accurate scene-level satellite alignment.

## 5 Conclusion

We introduced X-VGGT, a feed-forward model for georeferenced cross-view 3D reconstruction from ground and satellite images. X-VGGT jointly predicts ground-view cameras, dense 3D point maps, and a scene-level georeferencing transform that aligns the reconstruction with an orthorectified satellite image, enabling both camera geo-localization and dense pixel-wise geo-registration. We also proposed training and evaluation protocols covering ordered and unordered ground-view inputs across multiple datasets and sensor configurations. Experiments show that X-VGGT outperforms cross-view localization and 3D reconstruction baselines while preserving strong local 3D reconstruction quality, suggesting that robust georeferenced reconstruction requires coupling local multi-view geometry with explicit satellite-frame alignment.

## Acknowledgments

Supported by Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior/Interior Business Center (DOI/IBC) contract number 140D0423C0074. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DOI/IBC, or the U.S. Government.

## References

[1] Amar Ali-Bey, Brahim Chaib-Draa, and Philippe Giguere. Gsv-cities: Toward appropriate supervised visual place recognition. *Neurocomputing*, 513:194–203, 2022.

[2] Amar Ali-Bey, Brahim Chaib-Draa, and Philippe Giguere. Mixvpr: Feature mixing for visual place recognition. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 2998–3007, 2023.

[3] Relja Arandjelovic, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Netvlad: Cnn architecture for weakly supervised place recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5297–5307, 2016.

[4] Guillaume Astruc, Nicolas Dufour, Ioannis Siglidis, Constantin Aronssohn, Nacim Bouia, Stephanie Fu, Romain Loiseau, Van Nguyen Nguyen, Charles Raude, Elliot Vincent, Lintao Xu, Hongyu Zhou, and Loic Landrieu. Openstreetview-5m: The many roads to global visual geolocation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21967–21977, 2024.

[5] Guillaume Astruc, Eduard Trulls, Jan Hosang, Loic Landrieu, and Paul-Edouard Sarlin. Unigeoclip: Unified geospatial contrastive learning. *arXiv preprint arXiv:2604.11668*, 2026.

[6] Gabriele Berton, Carlo Masone, and Barbara Caputo. Rethinking visual geo-localization for large-scale applications. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4878–4888, 2022.

[7] Gabriele Berton, Gabriele Trivigno, Barbara Caputo, and Carlo Masone. Eigenplaces: Training viewpoint robust models for visual place recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11080–11090, 2023.

[8] Myron Brown, Michael Chan, and Michael Twardowski. Wriva public data, 2024.

[9] Yohann Cabon, Lucas Stoffl, Leonid Antsfeld, Gabriela Csurka, Boris Chidlovskii, Jerome Revaud, and Vincent Leroy. Must3r: Multi-view network for stereo 3d reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1050–1060, 2025.

[10] Brandon Clark, Alec Kerrigan, Parth Parag Kulkarni, Vicente Vivanco Cepeda, and Mubarak Shah. Where we are and what we're looking at: Query based worldwide image geo-localization using hierarchies and scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23182–23190, 2023.

[11] Fabian Deuser, Konrad Habel, and Norbert Oswald. Sample4geo: Hard negative sampling for cross-view geo-localisation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16847–16856, 2023.

[12] Siyan Dong, Shuzhe Wang, Shaohui Liu, Lulu Cai, Qingnan Fan, Juho Kannala, and Yanchao Yang. Reloc3r: Large-scale training of relative camera pose regression for generalizable, fast, and accurate visual localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16739–16752, 2025.

[13] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[14] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The international journal of robotics research*, 32(11):1231–1237, 2013.

[15] Lukas Haas, Michal Skreta, Silas Alberti, and Chelsea Finn. Pigeon: Predicting image geolocations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12893–12902, 2024.

[16] Kuan Wei Huang, Brandon Li, Bharath Hariharan, and Noah Snavely. C3po: Cross-view cross-modality correspondence by pointmap prediction. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2025.

[17] Nikhil Keetha, Avneesh Mishra, Jay Karhade, Krishna Murthy Jatavallabhula, Sebastian Scherer, Madhava Krishna, and Sourav Garg. Anyloc: Towards universal visual place recognition. *IEEE Robotics and Automation Letters*, 9(2):1286–1293, 2023.

[18] Nikhil Keetha, Norman Müller, Johannes Schönberger, Lorenzo Porzi, Yuchen Zhang, Tobias Fischer, Arno Knapitsch, Duncan Zauss, Ethan Weber, Nelson Antunes, Jonathon Luiten, Manuel Lopez-Antequera, Samuel Rota Bulò, Christian Richardt, Deva Ramanan, Sebastian Scherer, and Peter Kontschieder. MapAnything: Universal feed-forward metric 3D reconstruction. In *International Conference on 3D Vision (3DV)*. IEEE, 2026.

[19] Konstantin Klemmer, Esther Rolf, Caleb Robinson, Lester Mackey, and Marc Rußwurm. Satclip: Global, general-purpose location embeddings with satellite imagery. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 4347–4355, 2025.

[20] Ayesh Abu Lehyeh, Xiaohan Zhang, Ahmad Arrabi, Waqas Sultani, Chen Chen, and Safwan Wshah. Geoflow: Real-time fine-grained cross-view geolocalization via iterative flow prediction. *arXiv preprint arXiv:2603.21943*, 2026.

[21] Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding image matching in 3d with mast3r. In *European conference on computer vision*, pages 71–91. Springer, 2024.

[22] Guopeng Li, Ming Qian, and Gui-Song Xia. Unleashing unlabeled data: A paradigm for cross-view geo-localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16719–16729, 2024.

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

[24] Haotong Lin, Sili Chen, Junhao Liew, Donny Y Chen, Zhenyu Li, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth anything 3: Recovering the visual space from any views. *arXiv preprint arXiv:2511.10647*, 2025.

[25] Qingyang Liu, David G Shatwell, Parth Parag Kulkarni, and Mubarak Shah. Pixel-wise geo-registration of drone images. In *European Conference on Computer Vision*, pages 76–94. Springer, 2026.

[26] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[27] Eric Muller-Budack, Kader Pustu-Iren, and Ralph Ewerth. Geolocation estimation of photos using a hierarchical model and scene classification. In *Proceedings of the European conference on computer vision (ECCV)*, pages 563–579, 2018.

[28] 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.

[29] Manu S Pillai, Mamshad Nayeem Rizve, and Mubarak Shah. Garet: cross-view video geolocalization with adapters and auto-regressive transformers. In *European Conference on Computer Vision*, pages 466–483. Springer, 2024.

[30] Shraman Pramanick, Ewa M Nowara, Joshua Gleason, Carlos D Castillo, and Rama Chellappa. Where in the world is this image? transformer-based geo-localization in the wild. In *European Conference on Computer Vision*, pages 196–215. Springer, 2022.

[31] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 12159–12168. IEEE, 2021.

[32] Paul Hongsuck Seo, Tobias Weyand, Jack Sim, and Bohyung Han. Cplanet: Enhancing image geolocalization by combinatorial partitioning of maps. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 536–551, 2018.

[33] David G Shatwell, Ishan Rajendrakumar Dave, Sirnam Swetha, and Mubarak Shah. Gt-loc: Unifying when and where in images through a joint embedding space. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1–11, 2025.

[34] David G Shatwell, Sirnam Swetha, and Mubarak Shah. Tiger: A unified framework for time, images and geo-location retrieval. *arXiv preprint arXiv:2603.24749*, 2026.

[35] Yujiao Shi and Hongdong Li. Beyond cross-view image retrieval: Highly accurate vehicle localization using satellite image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17010–17020, 2022.

[36] Yujiao Shi, Hongdong Li, Akhil Perincherry, and Ankit Vora. Weakly-supervised camera localization by ground-to-satellite image registration. In *European Conference on Computer Vision*, pages 39–57. Springer, 2024.

[37] Zhenbo Song, Jianfeng Lu, Yujiao Shi, et al. Learning dense flow field for highly-accurate cross-view camera localization. *Advances in Neural Information Processing Systems*, 36:70612–70625, 2023.

[38] Thomas Sugg, Kyle O'Brien, Lekh Poudel, Alex Dumouchelle, Michelle Jou, Marc Bosch, Deva Ramanan, Srinivasa Narasimhan, and Shubham Tulsiani. Accenture-nvs1: A novel view synthesis dataset. *arXiv preprint arXiv:2503.18711*, 2025.

[39] Akihiko Torii, Josef Sivic, Tomas Pajdla, and Masatoshi Okutomi. Visual place recognition with repetitive structures. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 883–890, 2013.

[40] Issar Tzachor, Boaz Lerner, Matan Levy, Michael Green, Tal Berkovitz Shalev, Gavriel Habib, Dvir Samuel, Noam Korngut Zailer, Or Shimshi, Nir Darshan, et al. Effovpr: Effective foundation model utilization for visual place recognition. *arXiv preprint arXiv:2405.18065*, 2024.

[41] Vicente Vivanco Cepeda, Gaurav Kumar Nayak, and Mubarak Shah. Geoclip: Clip-inspired alignment between locations and images for effective worldwide geo-localization. *Advances in Neural Information Processing Systems*, 36:8690–8701, 2023.

[42] Nam Vo, Nathan Jacobs, and James Hays. Revisiting im2gps in the deep learning era. In *Proceedings of the IEEE international conference on computer vision*, pages 2621–2630, 2017.

[43] Khiem Vuong, Anurag Ghosh, Deva Ramanan, Srinivasa Narasimhan, and Shubham Tulsiani. Aerialmegadepth: Learning aerial-ground reconstruction and view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21674–21684, 2025.

[44] Shruti Vyas, Chen Chen, and Mubarak Shah. Gama: Cross-view video geo-localization. In *European Conference on Computer Vision*, pages 440–456. Springer, 2022.

[45] Jianyuan Wang, Nikita Karaev, Christian Rupprecht, and David Novotny. Vggsfm: Visual geometry grounded deep structure from motion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 21686–21697, 2024.

[46] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 5294–5306, 2025.

[47] Qiwei Wang, Shaoxun Wu, and Yujiao Shi. Bevsplat: Resolving height ambiguity via feature-based gaussian primitives for weakly-supervised cross-view localization. *arXiv preprint arXiv:2502.09080*, 2025.

[48] Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A. Efros, and Angjoo Kanazawa. Continuous 3d perception model with persistent state. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10510–10522, 2025.

[49] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20697–20709, 2024.

[50] Xiaolong Wang, Runsen Xu, Zhuofan Cui, Zeyu Wan, and Yu Zhang. Fine-grained cross-view geo-localization using a correlation-aware homography estimator. *Advances in Neural Information Processing Systems*, 36:5301–5319, 2023.

[51] Yifan Wang, Jianjun Zhou, Haoyi Zhu, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Jiangmiao Pang, Chunhua Shen, and Tong He. $\pi^3$: Permutation-equivariant visual geometry learning. *arXiv preprint arXiv:2507.13347*, 2025.

[52] Zengyan Wang, Sirshapan Mitra, Rajat Modi, Grace Lim, and Yogesh Rawat. Sky2ground: A benchmark for site modeling under varying altitude. *arXiv preprint arXiv:2603.13740*, 2026.

[53] Tobias Weyand, Ilya Kostrikov, and James Philbin. Planet-photo geolocation with convolutional neural networks. In *European conference on computer vision*, pages 37–55. Springer, 2016.

[54] Alexander Wollam, Kyle Ashley, Maxim Shugaev, Oliver Arend, Ilya Semenov, Hadis Dashtestani, Sumved Ravi, and Nathan Jacobs. Towards unconstrained cross-view pose estimation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 8448–8457, 2026.

[55] Zimin Xia and Alexandre Alahi. Fgˆ 2: Fine-grained cross-view localization by fine-grained feature matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6362–6372, 2025.

[56] Zimin Xia, Chenghao Xu, and Alexandre Alahi. Locˆ 2: Interpretable cross-view localization via depth-lifted local feature matching. In *The Fourteenth International Conference on Learning Representations*, 2025.

[57] Jianing Yang, Alexander Sax, Kevin J. Liang, Mikael Henaff, Hao Tang, Ang Cao, Joyce Chai, Franziska Meier, and Matt Feiszli. Fast3r: Towards 3d reconstruction of 1000+ images in one forward pass. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21924–21935, 2025.

[58] Junyan Ye, Zhutao Lv, Weijia Li, Jinhua Yu, Haote Yang, Huaping Zhong, and Conghui He. Cross-view image geo-localization with panorama-bev co-retrieval network. In *European Conference on Computer Vision*, pages 74–90. Springer, 2024.

[59] Junyi Zhang, Charles Herrmann, Junhwa Hur, Varun Jampani, trevor darrell, Forrester Cole, Deqing Sun, and Ming-Hsuan Yang. Monst3r: A simple approach for estimating geometry in the presence of motion. In *International Conference on Learning Representations*, pages 82863–82886, 2025.

[60] Shangzhan Zhang, Jianyuan Wang, Yinghao Xu, Nan Xue, Christian Rupprecht, Xiaowei Zhou, Yujun Shen, and Gordon Wetzstein. Flare: Feed-forward geometry, appearance and camera estimation from uncalibrated sparse views. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21936–21947, 2025.

[61] Xiaohan Zhang, Waqas Sultani, and Safwan Wshah. Cross-view image sequence geo-localization. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2914–2923, 2023.

[62] Sijie Zhu, Taojiannan Yang, and Chen Chen. Vigor: Cross-view image geo-localization beyond one-to-one retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3640–3649, 2021.

[63] Sijie Zhu, Mubarak Shah, and Chen Chen. Transgeo: Transformer is all you need for cross-view image geo-localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1162–1171, 2022.

# Appendix

## A Limitations

X-VGGT assumes access to an orthorectified, georeferenced satellite image with known spatial resolution. Performance may degrade when the satellite image is outdated, low resolution, occluded, or visually inconsistent with the ground views, since the model relies on cross-view evidence to align the local reconstruction to the satellite frame.

The model is also trained with satellite crops in a specific physical scale range, roughly $150$–$250$ meters across depending on the dataset. Applying X-VGGT to substantially smaller or larger satellite regions may change the apparent scale and context seen by the model, and would likely require fine-tuning or adapting the crop protocol.

## B Broader Impacts

This work studies georeferenced 3D reconstruction from ground-level and satellite images, unifying cross-view geo-localization with 3D reconstruction. We hope it can serve as an initial step toward treating these two tasks jointly. Potential positive impacts include improved mapping from ground images, localization in GPS-denied or GPS-degraded environments, disaster response using localized post-event images, infrastructure inspection, and robotics or autonomous navigation systems that need to connect local observations to a geographic map.

As with geo-localization methods more broadly, responsible use should consider privacy and data governance when applying the model to images of real-world places. The method should be used in accordance with dataset licenses, local regulations, and appropriate safeguards for images that may contain sensitive locations, private property, people, or vehicles. Since georeferencing errors can occur, predictions should be validated before use in safety-critical or decision-making settings.

## C Additional Training Details

### C.1 Training Satellite Images

X-VGGT is trained on Sat-MegaDepth and SatKITTI. Sat-MegaDepth extends Aerial-MegaDepth with one orthorectified, georeferenced satellite mosaic per scene. It contains 121 scenes and 63,616 ground images, divided into 116 training scenes with 60,768 images and five validation scenes with 2,848 images. Table 6 summarizes the physical coverage and native spatial resolution of the satellite mosaics across both splits.

Each training sample contains a set of ground images, a square satellite crop, camera calibration, ground-truth 3D points, and satellite-coordinate targets for the selected ground cameras. We sample satellite crops spanning 200–250 m per side for Sat-MegaDepth and 150–250 m for SatKITTI, then resize them to $518 \times 518$ pixels. These crops provide different satellite contexts within the larger scene mosaics. Figure 5 shows an example from Sat-MegaDepth.

Table 6: **Sat-MegaDepth satellite imagery statistics.** Statistics are computed across the combined training and validation sets, comprising 121 satellite mosaics, before crop sampling or resizing. Physical dimensions are measured from the georeferenced image corners, and area is approximated as width $\times$ height.

| Measurement | Minimum | Mean | Median | Maximum |
|---|---|---|---|---|
| Width (m) | 502.2 | 748.7 | 617.8 | 1,649.6 |
| Height (m) | 502.2 | 760.9 | 627.4 | 2,018.6 |
| Area ($km^2$) | 0.263 | 0.628 | 0.390 | 2.206 |
| Native resolution (cm/pixel) | 7.42 | 10.41 | 10.46 | 12.99 |



Figure 5: **Example from the Sat-MegaDepth split.** The ground images at both sides are taken from the original MegaDepth [23] dataset. We obtain high-resolution orthorectified satellite images covering the location of all ground views from Google Maps.

### C.2 Coordinate Targets and Augmentation

The satellite supervision uses normalized $[0, 1]^2$ coordinates with a vertical axis convention in which larger values correspond to upward displacements in the satellite plane. Each dataloader returns normalized camera-center targets $\mathbf{u}_i^{\mathrm{cam}}$ and the satellite image scale in meters. During training, the satellite image is augmented with horizontal flips and random $90^\circ$ rotations. The same transform is applied to satellite-coordinate targets, the ENU world frame, camera extrinsics, and point maps so that the georeferencing supervision remains consistent after augmentation.

Before computing the losses, we apply the VGGT-style normalization of camera extrinsics and point maps into the local reconstruction frame. Because this normalization changes the coordinate frame in which the model predicts point maps and camera poses, we recompute the ground-truth georeferencing parameters $\phi^*$, dense satellite targets, and valid dense projection masks after normalization. This recomputation keeps $\mathcal{G}_{\phi^*}$, camera-center targets, and dense point targets in the same coordinate frame as the model predictions.

### C.3 Optimization

The model is initialized from the pretrained VGGT checkpoint. New satellite camera/register tokens and the Satellite Alignment Head (SAH) are optimized from the start of training to learn the cross-view alignment needed for georeferencing, while the camera and point heads retain their reconstruction roles. The top four global attention blocks are unfrozen after $10\%$ of training, while the camera head, point head, and remaining default parameters are unfrozen after $40\%$ of training. We use separate warmup and cosine schedules for each parameter group, with the largest learning rate for the SAH and smaller learning rates for the pretrained backbone and 3D heads. Training batches contain 4–32 ground images per scene; all images are resized to $518 \times 518$ pixels, with zero-padding for rectangular aspect ratios. We use the AdamW optimizer [26] with weight decay $0.05$, bfloat16 automatic mixed precision, gradient clipping, and train for 80,000 iterations on four H100 GPUs. Models trained for the georeferencing loss ablation in Table 5 use the same training schedule, but are trained for 10,000 iterations on two H100 GPUs.

# D Additional Evaluation Details

## D.1 Evaluation datasets.

**KITTI** [14] is an autonomous driving dataset collected around Karlsruhe, Germany, including urban roads, rural areas, and highways. The capture platform is equipped with four cameras, two grayscale and two RGB, a Velodyne LiDAR scanner, and a high-precision GPS/IMU system. We use the augmented benchmark of Shi and Li [35], which provides satellite images around the vehicle trajectory. For our evaluation, we use only the left RGB camera: the left and right stereo views are highly similar, and our goal is to evaluate sequence-level ground-satellite georeferencing rather than stereo reconstruction. We use LiDAR-derived depth maps for pixel-wise evaluation and report results on the challenging cross-area Test-2 split.

**Mapillary Metropolis** is a city-scale driving dataset collected in Chicago. It provides synchronized camera poses, georeferenced aerial imagery, panoramic street-level images, and four perspective images at each timestamp, corresponding to the front, back, left, and right directions relative to the vehicle. The dataset also includes multi-view-stereo reconstructions, MVS-derived depth maps for the perspective images, and LiDAR point clouds. Although LiDAR points are available, they are sparser than the MVS-derived depth maps for dense image evaluation, so we use the MVS depth maps when computing pixel-wise geo-registration metrics. We use Metropolis for zero-shot evaluation, since it differs from our training data in sensor layout, capture protocol, and scene distribution.

**WRIVA** is derived from the outdoor sites of the WRIVA Public Dataset [8]. Unlike KITTI and Metropolis, which are vehicle-centric driving datasets, WRIVA contains unstructured image collections from six U.S. sites with cameras scattered around each scene. The scenes have substantially different layouts from urban driving datasets, consisting largely of rural environments, open areas, and training-ground-style sites relevant to first-responder and site-modeling scenarios. WRIVA provides calibrated camera parameters and GPS annotations. Although the original dataset includes low-resolution panchromatic and multispectral satellite imagery, these modalities are outside the scope of our work, so we replace them with high-resolution orthorectified RGB satellite images from Google Maps. Since WRIVA does not provide ground-truth depth maps, we use it only for camera geo-localization and pose evaluation.

**ACC-NVS1** [38] is a multi-view dataset collected for novel view synthesis across six real-world scenes in Austin, TX and Pittsburgh, PA. It contains approximately 148K images captured from both ground and airborne viewpoints, with calibrated camera intrinsics, extrinsics, and geolocation. In our evaluation, we use the ground-level image sequences and supplement them with high-resolution orthorectified RGB satellite images to support cross-view geo-localization. Since ACC-NVS1 does not provide ground-truth depth maps, we use it only for camera geo-localization and pose evaluation.

## D.2 Evaluation scene generation.

We construct evaluation scenes using dataset-specific sampling protocols that reflect the structure of each dataset while keeping the evaluation format consistent across benchmarks.

**KITTI** scenes are generated from linear driving sequences. We sample contiguous or nearby frames from each sequence to form ordered ground-view sets, using only the left RGB camera. This produces sequence-like scenes where the model must estimate a shared satellite alignment from multiple views along the vehicle trajectory.

**Mapillary Metropolis** scenes are constructed from consecutive capture groups. Each capture group contains four perspective images aligned with the vehicle frame, corresponding to orthogonal viewing directions. We preserve this multi-directional structure when creating the scenes, which provides denser coverage around the vehicle than a single forward-facing driving sequence.

**WRIVA** contains unordered image collections rather than vehicle trajectories. We follow a VGGT-style batch construction strategy: for each scene, we choose a seed image and sample additional views with high pairwise pose similarity. This produces coherent local scenes while preserving the unordered nature of the dataset.

**ACC-NVS1** scenes are generated from ground-level image sequences. Similar to KITTI, we sample ordered frame sets from each sequence to evaluate camera geo-localization under trajectory-like motion, but on locations and capture conditions not seen during training.

**Evaluation Satellite Images.** For every evaluation scene, we compute the centroid of the selected camera centers and extract a $150\text{m} \times 150\text{m}$ orthorectified satellite crop. Rather than placing the centroid exactly at the crop center, we randomly position it within the central $100\text{m} \times 100\text{m}$ region. This prevents the evaluation from assuming a perfectly centered satellite crop while ensuring that the selected cameras remain inside the image. For the main results, we sample up to $1024$ scenes per dataset with $16$ ground images per scene. For the ablation on the number of ground images, we construct larger scenes with up to $96$ ground images.

## E Metric Depth Estimation

The known ground sampling distance of the satellite image provides a metric reference for the reconstructed scene, allowing X-VGGT to resolve the scale ambiguity in the ground 3D reconstruction and generate metric depth maps. The Satellite Alignment Head predicts a log-scale $\alpha$ that maps reconstruction units to normalized satellite coordinates. For a square satellite crop spanning $\rho_{\text{sat}}$ meters per side, the corresponding metric scale is $\rho_{\text{sat}} \exp(\alpha)$. We transform each predicted point into its ground camera's coordinate frame and extract its depth:

$$\hat{d}_i(p) = \rho_{\text{sat}} \exp(\alpha) \left[\hat{R}_i \hat{\mathbf{X}}_i(p) + \hat{\mathbf{t}}_i\right]_z, \tag{1}$$

where $\hat{\mathbf{X}}_i(p)$ is the predicted 3D point at pixel $p$, $[\hat{R}_i \mid \hat{\mathbf{t}}_i]$ is the predicted camera pose w.r.t. the reference camera, and $[\cdot]_z$ selects the camera-frame depth component. Figure 6 compares the ground truth and predicted depth maps, showing that X-VGGT is able to successfully recover the metric scale from the georeferenced satellite image.

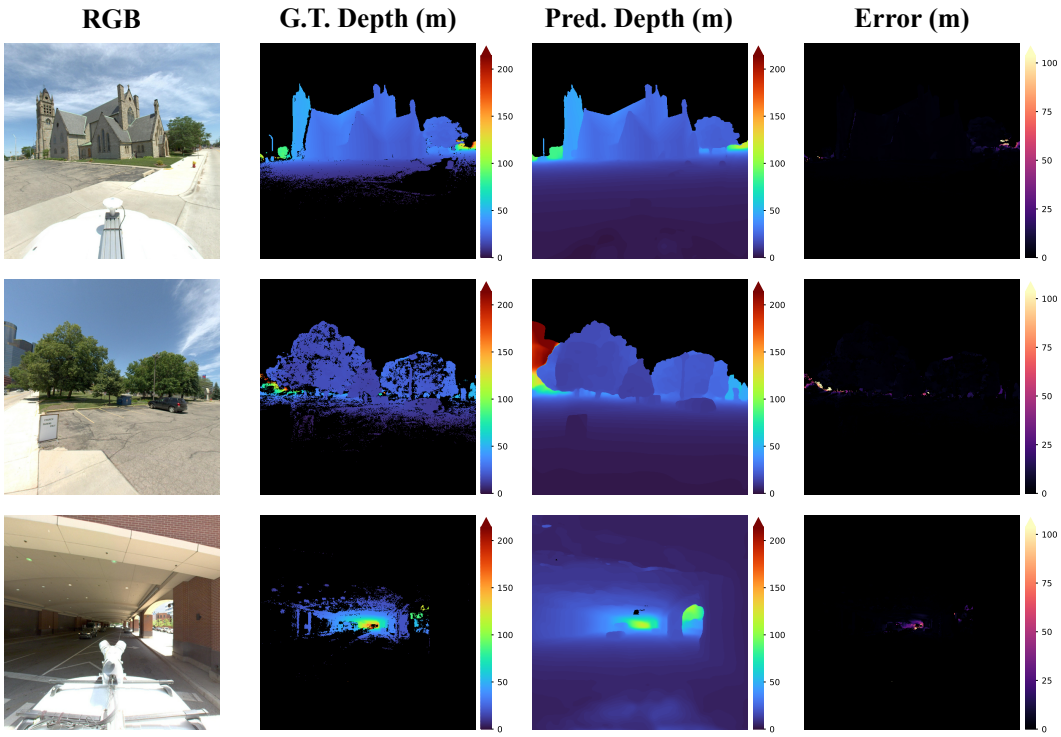


Figure 6: **Qualitative metric depth predictions.** From left to right: input ground image, ground-truth depth, X-VGGT metric depth, and absolute depth error. Ground-truth and predicted depth maps share the same colormap and range across all examples; error maps use a separate shared scale. All values are in meters. Missing ground-truth depth is shown in black, and errors are computed only where ground-truth depth is available.

## F Effect of Ground–Satellite Interaction

To study the role of joint cross-view encoding, we compare X-VGGT with a late-fusion baseline that processes the ground views and satellite image in separate passes through a shared encoder. The ground views retain token-level interaction with one another, but ground and satellite tokens do not interact directly. The alignment head processes the two streams separately, pools the ground camera tokens, and combines the ground and satellite summaries with an MLP to predict the georeferencing transform. The camera and point heads receive only ground-view features.

Table 7 shows that late fusion increases camera localization, orientation, and dense registration errors on both KITTI and Metropolis. For example, KITTI localization error rises from $24.62$ m to $43.58$ m, while orientation error rises from $22.50°$ to $95.07°$. These results support the value of jointly encoding ground and satellite views for estimating the scene-level alignment.

Table 7: **Effect of ground–satellite interaction.** Mean camera localization error (Loc.), orientation error (Orien.), and pixel-wise geo-registration error (G.R.) for the late-fusion baseline and X-VGGT. Both methods use 16 ground images and one satellite image. Lower is better.

| Method | KITTI | | | Metropolis | | |
|---|---|---|---|---|---|---|
| | Loc. (m) | Orien. (°) | G.R. (m) | Loc. (m) | Orien. (°) | G.R. (m) |
| Late fusion | 43.58 | 95.07 | 44.21 | 41.96 | 87.26 | 54.81 |
| **X-VGGT** | **24.62** | **22.50** | **25.52** | **14.35** | **17.59** | **17.84** |

## G Inference Efficiency

We measure X-VGGT's inference latency and peak GPU memory as the number of ground images increases from 4 to 64. Experiments use an NVIDIA A100 80GB GPU, batch size of 1 scene, image resolution of $518 \times 518$ pixels, and FP16 mixed precision. After 10 warm-up iterations, we report the median latency and peak PyTorch-allocated memory over 30 forward passes, synchronizing the GPU for timing. Latency excludes model loading, data loading, preprocessing, and CPU-to-GPU tensor transfer.

Table 8: **Inference efficiency.** Inference latency and peak GPU memory of X-VGGT for different numbers of ground images. Every setting additionally includes one satellite image.

| Ground images | Latency (s) | Peak GPU memory (GiB) |
|---|---|---|
| 4 | 0.244 | 9.30 |
| 8 | 0.470 | 11.50 |
| 16 | 1.020 | 13.63 |
| 32 | 2.579 | 17.90 |
| 64 | 7.539 | 26.65 |

Table 8 shows that the 16-ground-view setting used in the main experiments requires $1.020$ s and $13.63$ GiB of GPU memory. Increasing the number of views raises both computational cost and memory usage, with latency growing more rapidly at larger view counts. These measurements complement the view-count accuracy ablation in Figure 4 by quantifying the cost of incorporating additional views.

## H Robustness to Satellite Configurations

X-VGGT uses satellite imagery to place reconstructed cameras and 3D points in a geographic coordinate frame. We examine how this alignment is affected by the area shown in the satellite image, the amount of visual detail it contains, and errors in its geographic coordinates. On KITTI and Metropolis, we test these factors separately by changing the spatial dimensions, reducing the resolution, and introducing spatial offsets, while keeping each scene's 16 ground images fixed. To

compare all settings on the same scenes, we require the cameras to remain visible within the smallest satellite crop and sufficient surrounding imagery to extract larger or shifted crops without padding. Specifically, all camera centers must lie at least 5 m inside the smallest crop, and the available imagery must accommodate the largest crop and shifts of up to 10 m. We select 512 scenes per dataset that satisfy these requirements using a fixed seed of 0, before inference and independently of model predictions.

These requirements exclude scenes with cameras spread over a large area or positioned near the satellite boundary. Our main evaluation instead uses an independent selection of 1,024 scenes per dataset without these restrictions. The two evaluations therefore contain different scenes, which must be considered when interpreting their results—for example, the unmodified satellite images yield a KITTI localization error of $12.88$ m here, compared with $24.62$ m in Table 2. To measure the effect of each change, we compare it against the unmodified satellite image using exactly the same 512 scenes.

**Satellite crop size.** We vary the width and height of the satellite crop by $\pm 25\%$ around the standard $150 \times 150$ m region, keeping the crop center and ground images fixed. Each crop is resized to $518 \times 518$ pixels, and we update its geographic bounds and the corresponding evaluation targets. Table 9 shows that larger crops increase localization and geo-registration errors on both datasets. Smaller crops yield similar KITTI localization error and lower Metropolis errors, although KITTI orientation error increases.

Table 9: **Robustness to satellite crop size.** Mean camera localization error (Loc.), orientation error (Orien.), and pixel-wise geo-registration error (G.R.) on the same 512 scenes per dataset. Crop size denotes the side length of the square satellite region; all crops are resized to $518 \times 518$ pixels. Lower is better.

| Crop size | KITTI | | | Metropolis | | |
|---|---|---|---|---|---|---|
| | Loc. (m) | Orien. (°) | G.R. (m) | Loc. (m) | Orien. (°) | G.R. (m) |
| 112.5 m (0.75×) | 12.78 | 17.55 | 13.85 | 9.36 | 12.99 | 11.81 |
| 150 m (1×) | 12.88 | 14.71 | 13.17 | 10.53 | 13.36 | 13.77 |
| 187.5 m (1.25×) | 17.06 | 15.42 | 15.78 | 13.27 | 13.89 | 17.65 |

**Satellite resolution.** We evaluate the robustness of X-VGGT to different satellite resolutions by downsampling the $518 \times 518$ satellite image to $259 \times 259$ or $130 \times 130$ pixels, then resize it back to $518 \times 518$ using bicubic interpolation. The ground images, geographic bounds, and evaluation targets remain fixed. Table 10 shows increasing errors as satellite detail is reduced. At $130 \times 130$ pixels, localization error rises from $12.88$ m to $25.28$ m on KITTI and from $10.53$ m to $20.04$ m on Metropolis, highlighting the importance of spatial detail for cross-view alignment.

Table 10: **Robustness to satellite resolution.** Satellite images are downsampled to the listed resolution and then resized back to the model's $518 \times 518$ input resolution, without changing the geographic region. Metrics and scenes are the same as in Table 9. Lower is better.

| Resolution | KITTI | | | Metropolis | | |
|---|---|---|---|---|---|---|
| | Loc. (m) | Orien. (°) | G.R. (m) | Loc. (m) | Orien. (°) | G.R. (m) |
| $518 \times 518$ (1.00×) | 12.88 | 14.71 | 13.17 | 10.53 | 13.36 | 13.77 |
| $259 \times 259$ (0.50×) | 19.90 | 17.78 | 18.25 | 14.16 | 13.81 | 17.37 |
| $130 \times 130$ (0.25×) | 25.28 | 24.68 | 23.88 | 20.04 | 22.49 | 25.57 |

**Satellite misalignment.** To simulate errors in satellite georeferencing, we randomly shift the satellite images by 5 or 10 m in any cardinal direction, while retaining the original geographic bounds and evaluation targets. This introduces a mismatch between the image content and its assigned geographic coordinates. Table 11 shows that these shifts primarily affect localization and geo-registration, while orientation errors remain comparatively stable. With a 10 m shift, localization error increases from $12.88$ m to $16.74$ m on KITTI and from $10.53$ m to $14.80$ m on Metropolis.

Table 11: **Robustness to satellite misalignment.** The satellite image is shifted by the listed distance while its assigned geographic bounds and evaluation targets are held fixed. Shift directions are balanced across north, east, south, and west. Metrics and scenes are the same as in Table 9. Lower is better.

| Image shift | KITTI | | | Metropolis | | |
|---|---|---|---|---|---|---|
| | Loc. (m) | Orien. (°) | G.R. (m) | Loc. (m) | Orien. (°) | G.R. (m) |
| 0 m | 12.88 | 14.71 | 13.17 | 10.53 | 13.36 | 13.77 |
| 5 m | 14.27 | 14.80 | 14.50 | 11.96 | 13.08 | 15.09 |
| 10 m | 16.74 | 15.36 | 17.05 | 14.80 | 13.29 | 17.86 |

# I Oracle Alignment

The geographic accuracy of X-VGGT depends on both the reconstructed geometry and its alignment with the satellite image. To examine how much error can be reduced by improving alignment, we compare the predicted alignment with an oracle obtained from ground-truth camera locations. For each scene, we fit a single Sim(3) transform, consisting of a rotation, translation, and uniform scale, between X-VGGT's 16 predicted camera centers and their ground-truth 3D locations. We apply this transform to all predicted cameras and dense points, then project them into the satellite plane using the ground-truth satellite transform.

Table 12 shows that oracle alignment substantially reduces camera localization and orientation errors on both datasets. On KITTI, localization error decreases from $24.62$ m to $0.09$ m, while pixel-wise geo-registration error decreases from $25.52$ m to $7.65$ m, indicating considerable room to improve the predicted alignment. On Metropolis, camera localization improves from $14.35$ m to $1.88$ m and orientation error decreases from $17.59°$ to $3.78°$, but dense geo-registration improves only from $17.84$ m to $16.84$ m. Thus, accurately aligning camera centers does not by itself ensure accurate placement of the dense reconstruction.

Table 12: **Oracle alignment comparison.** Mean camera localization error (Loc.), orientation error (Orien.), and pixel-wise geo-registration error (G.R.) for X-VGGT using its predicted alignment or a scene-level alignment fitted to ground-truth camera centers. Both settings use the same predicted local reconstruction. Lower is better.

| Method | KITTI | | | Metropolis | | |
|---|---|---|---|---|---|---|
| | Loc. (m) | Orien. (°) | G.R. (m) | Loc. (m) | Orien. (°) | G.R. (m) |
| X-VGGT | 24.62 | 22.50 | 25.52 | 14.35 | 17.59 | 17.84 |
| X-VGGT + oracle alignment | 0.09 | 6.49 | 7.65 | 1.88 | 3.78 | 16.84 |

# J Additional Qualitative Examples

Figure 7 shows additional X-VGGT prediction examples across evaluation scenes. Figure 8 compares X-VGGT against baselines on camera geo-localization and shows that our model produces more accurate predictions across all datasets.

# K Licenses and Existing Assets

We use existing datasets and pretrained models, and cite the original sources throughout the paper. KITTI and Mapillary Metropolis are released under a Creative Commons Attribution-NonCommercial-ShareAlike license, and MegaDepth-derived data is released under a Creative Commons Attribution license. We use these assets for research purposes and include qualitative examples in accordance with the corresponding dataset licenses and terms. The paper is distributed under a license compatible with the reproduced third-party visual assets.




Figure 7: **Additional qualitative predictions.** We show more example predictions from X-VGGT across scenes from KITTI (first two rows) and Mapillary Metropolis (last two rows).




Figure 8: **Qualitative comparison with baselines.** We compare X-VGGT against the baselines on camera geo-localization across the four evaluation datasets. From top to bottom: KITTI, Mapillary Metropolis, WRIVA, and ACC-NVS1.

## NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main claims in the abstract and introduction are supported by the method and experiments: the paper introduces X-VGGT, a feed-forward model for jointly predicting camera poses, dense point maps, and a satellite alignment transform, and evaluates camera geo-localization, pixel geo-localization, and 3D reconstruction across multiple datasets.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses limitations in Appendix A, including the dependence on suitable orthorectified georeferenced satellite imagery and the fact that the model is trained on satellite crops within a specific physical scale range. It notes that substantially different satellite crop sizes may require fine-tuning or adapting the crop protocol.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: The paper does not include theoretical results.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper describes the model architecture, georeferencing losses, training datasets, data construction procedure, optimization setup, evaluation datasets, baselines, and metrics in Sections 3.2–3.5, the experiments section, and Appendix C. These details provide the information needed to reproduce the main experimental comparisons and evaluate the claims about camera geo-localization, pixel geo-localization, and 3D reconstruction.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: The code and processed datasets are not currently publicly available because the release package is still being prepared. We plan to release the code and instructions needed to reproduce the training and evaluation protocols.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The paper specifies the main training and evaluation settings, including the training datasets, supervision signals, image resolution, number of input ground views, optimizer, loss weights, training schedule, evaluation datasets, scene construction protocols, baselines, and metrics. These details are provided in Sections 3.5 and the experiments section, with additional crop sampling, augmentation, and optimization details in Appendix C.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

Justification: We report mean performance across evaluation scenes, but do not report error bars or statistical significance over multiple independently trained models because each full training run requires substantial compute.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper reports the main compute resources used for training, including the use of four H100 GPUs, 80,000 training iterations, bfloat16 mixed precision, and the optimizer/training configuration in Section 3.5. The experiments are feed-forward evaluations on the reported datasets, and the main computational cost is the training run.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have reviewed the NeurIPS Code of Ethics and are not aware of any aspect of the research that violates it. The work uses existing vision/geospatial datasets for model training and evaluation, and does not involve human-subject experiments or deployment in a decision-making system.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.

- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The paper discusses potential positive impacts in mapping, robotics, disaster response, and infrastructure inspection, and notes that responsible use of geo-localization methods should consider privacy, data governance, and validation before safety-critical deployment. These issues are discussed in Appendix B.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: We do not release any training dataset, and instead plan to release training and evaluation protocols using existing open source datasets. For any future release of model weights, we will apply anonymization filters to the training images where needed and retrain the final released model; we note that Mapillary Metropolis already blurs personal information such as faces and license plates, while KITTI and MegaDepth do not provide the same anonymization guarantees.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.

- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We cite the original datasets, models, and papers used in this work, and discuss the relevant licenses and terms for existing assets in Appendix K. The paper is distributed under a license compatible with the reproduced third-party visual assets.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: We do not release any new assets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.