This CVPR 2021 workshop paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Single View Geocentric Pose in the Wild

Gordon Christie[1], Kevin Foster[1], Shea Hagstrom[1], Gregory D. Hager[2], Myron Z. Brown[1]
[1]The Johns Hopkins University Applied Physics Laboratory
[2]Department of Computer Science, The Johns Hopkins University
{gordon.christie,kevin.foster,shea.hagstrom,myron.brown}@jhuapl.edu, hager@cs.jhu.edu

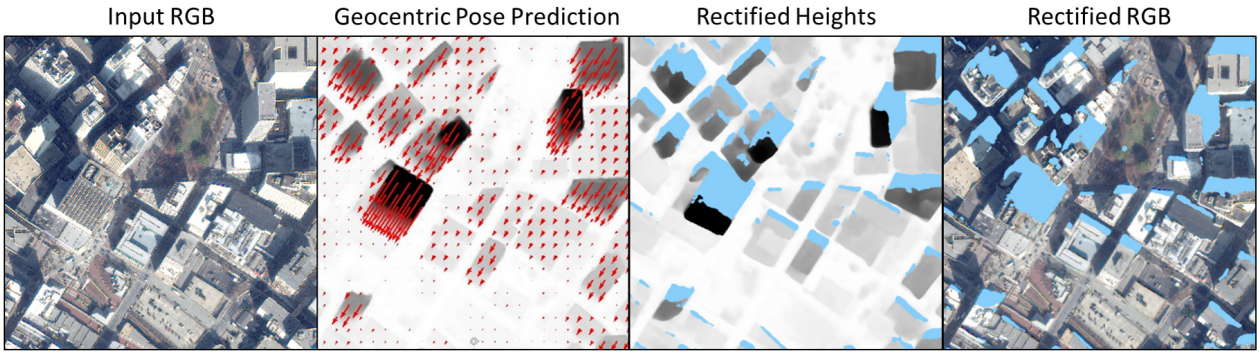

Figure 1: Our model predicts object heights and vector fields mapping surface features to ground level, enabling feature rectification and occlusion mapping. Darker shades of gray have larger height values, red arrows map surface features to ground level, and occluded pixels are blue.

Abstract

Current methods for Earth observation tasks such as semantic mapping, map alignment, and change detection rely on near-nadir images; however, often the first available images in response to dynamic world events such as natural disasters are oblique. These tasks are much more difficult for oblique images due to observed object parallax. There has been recent success in learning to regress an object's geocentric pose, defined as height above ground and orientation with respect to gravity, by training with airborne lidar registered to satellite images. We present a model for this novel task that exploits affine invariance properties to outperform state of the art performance by a wide margin. We also address practical issues required to deploy this method in the wild for real-world applications. Our data and code are publicly available [1].

1. Introduction

The ability to accurately estimate 3D scene geometry from a single satellite image can dramatically improve automated scene understanding for Earth observation tasks such as urban development monitoring and damage assessment after natural disasters. Most methods for feature mapping [8], map alignment [4], and change detection [9] require near-nadir images for good performance. Geospatially localizing features in more oblique images is challenging due to above-ground image parallax and occlusion [40]. These issues can be addressed explicitly with known 3D scene geometry, but such information is generally not known in advance. Until recently, methods for predicting this 3D geometry from a single satellite image also relied on near-nadir imaging geometry for good performance [38, 29, 27].

Christie et al. [6] recently proposed a method to address this challenge with oblique images by regressing geocentric pose, defined as an object's height above ground and orientation with respect to gravity [14]. Their method, supervised by lidar, represents geocentric pose with pixel-level object heights and vector fields mapping surface pixels to ground level. While they demonstrated promising initial results, their model fails to reliably predict heights for tall buildings. In this work, we present a solution that exploits affine invariances to outperform state of the art by a wide margin (Fig. 2). Our model produces accurate heights and vector fields even for very tall buildings and produces accurate occlusion maps (Fig. 1). We also explore practical issues required to deploy our method for real-world applications. Specifically, we make the following contributions:

[1] https://github.com/pubgeo/monocular-geocentric-pose

(1) We review affine imaging geometry and exploit invariances to explicitly model the relationship between object heights and the vector field that maps surface features to ground level in an image. (2) To improve prediction of taller building heights, we propose a novel strategy for fast augmentations to synthetically increase the heights of objects by inverting geocentric pose vector fields. (3) We outperform state of the art for height prediction [38, 29, 27, 19, 43] and geocentric pose [6] and demonstrate accurate predictions even for orthorectified images that violate our affine assumptions. (4) We present the first demonstration of supervising this task without lidar, using only geometry derived from images that can be produced anywhere on Earth. (5) We extend the public dataset from [6] to increase geographic diversity and produce consistent train and test sets for public leaderboard evaluation. We make our code available as a strong baseline to promote further research.

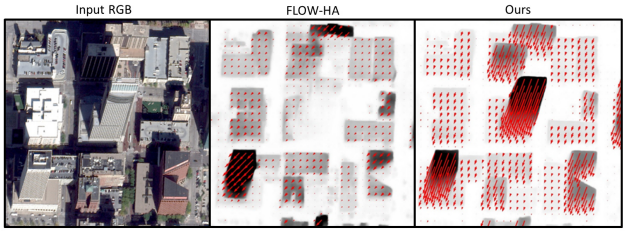

Figure 2: State of the art FLOW-HA from Christie et al. [6] under-predicts heights and vector field magnitudes for tall buildings. Our model and augmentations exploit affine invariances to produce accurate predictions. Darker gray is taller, and red arrows map surface features to ground level.

2. Related Work

2.1. Monocular Height Prediction

Deep networks for monocular depth prediction [12, 42, 13, 23] have been very successful for applications such as autonomous vehicles where the observed scene is tens of meters from the sensor. Inspired by these successes, recent work has demonstrated similar methods to predict height above ground for Earth observation images where the scene can be hundreds of kilometers from the sensor [29].

Height and semantic category are complementary intrinsic object attributes. Knowledge of semantic category can constrain height predictions. Trivial examples for remote sensing are ground and water that have no height. Other features such as trees and buildings have known distributions of physically plausible heights. Kunwar [19] and Zheng et al. [43] leveraged semantic cues as priors for height prediction to win the 2019 Data Fusion Contest (DFC19) single-view semantic 3D challenge track [20]. Srivistava et al. [38] proposed to learn semantics and height jointly with a multi-task deep network. Mahmud et al. [27] proposed an especially intriguing model that jointly learns to perform semantic segmentation, height above ground, and a signed distance function from building boundaries. This method in particular relies on near-nadir views where building footprints are consistent with observed roof appearance.

In our work, we explicitly account for oblique imaging geometry, making no assumption of near-nadir views. We do not explicitly reason about semantics, but we still find the height estimates from our geocentric pose model are more accurate than those produced by these state of the art height prediction models that do (Sec. 5.2). We expect that finer-grained semantic labels for buildings (e.g., commercial vs. residential) or other attributes such as building footprint size may provide more useful complementary information to improve our model. We leave these questions for future work.

2.2. Geocentric Pose

Geocentric pose, defined as height above ground and orientation with respect to gravity, was originally proposed by Gupta el al. [14] and used as a feature for object recognition and scene classification. They also proposed a three-channel geocentric pose representation called HHA that encodes horizontal disparity (equivalent to depth), height above ground, and orientation with respect to gravity [15]. This representation has since been used by many works focused on scene understanding [5, 32, 24, 31, 39, 26, 36, 25].

Inspired by the success of the this representation, Christie et al. [6] proposed learning geocentric pose for rectifying oblique monocular satellite images and produced the first public dataset for this task, including satellite images over three cities in the United States. Their model (Fig. 4) relates height as a prior for vector field magnitude without an adequate mechanism to learn the true relationship, resulting in poor estimates for tall buildings. In our work, we explicitly model this relationship based on the geometry of affine projection, resulting in more accurate predictions for tall objects. We are currently not aware of any other published work on this topic. To promote further research and enable a public leaderboard for evaluation, we extend the dataset from [6] to include a challenging city outside the United States and to address inconsistencies in the original dataset (Sec. 5.1).

3. Affine Geocentric Pose

Satellite pushbroom sensors are well-approximated locally (e.g., for processing image tiles) as affine cameras [7], so we define our geocentric pose representation explicitly for an affine camera. Depth variations for objects in the scene are much smaller than depth from the sensor, angular field of view is narrow for a local sub-image, and the sensor maintains approximately the same attitude and speed throughout image acquisition. Fig. 3 illustrates the invariant

properties of affine projection compared to the more general perspective projection model, as described in detail in [16]. We exploit the invariant property of parallelism to define a single angle of parallel projection θ for each image. Unlike [6], we also exploit preservation of the ratio of lengths on parallel lines to relate object heights with magnitudes of vectors mapping surface features to ground level.

We now define the mathematical assumptions in our model. Given image $\boldsymbol{I}$ with 2x3 affine projection matrix $\boldsymbol{A}$, we define geocentric pose $g(\boldsymbol{I}) = \{s, \theta, \boldsymbol{h}\}$, where the vector $\boldsymbol{h}$ is height above ground level (AGL) for each pixel in image $\boldsymbol{I}$, θ is the angle of parallel projection in the image plane, and s is the scale factor relating lengths of lines along those parallel projections in world coordinates and their corresponding lengths in image coordinates.

First we define affine projection $\boldsymbol{A}$ of any two vertically aligned world coordinates P_1 and P_2 to their image coordinates p_1 and p_2 observed in image $\boldsymbol{I}$.

$$p_1 = \begin{pmatrix} x_1 & y_1 \end{pmatrix}^T = \boldsymbol{A} P_1 \tag{1}$$

$$p_2 = \begin{pmatrix} x_2 & y_2 \end{pmatrix}^T = \boldsymbol{A} P_2 \tag{2}$$

$$P_1 = \begin{pmatrix} X & Y & Z_1 \end{pmatrix}^T \tag{3}$$

$$P_2 = \begin{pmatrix} X & Y & Z_2 \end{pmatrix}^T ; Z_2 > Z_1 \tag{4}$$

We then define height h as the distance between P_1 and P_2 (meters) and magnitude m as the distance between p_1 and p_2 (pixels).

$$h = Z_2 - Z_1 ; m = \|p_2 - p_1\| \tag{5}$$

Given these two image coordinates p_1 and p_2, height h (meters), and corresponding projected magnitude m (pixels), we determine the angle of parallel projection θ (radians) and scale factor s (pixels/meter). Observe that scale s is zero for local nadir imaging geometry.

$$\theta = atan2\left(y_2 - y_1, x_2 - x_1\right) ; s = m/h \tag{6}$$

Without loss of generality, we constrain all points $\boldsymbol{P_1}$ to be at ground level and we define the vector field $\dot{\boldsymbol{p}}$ mapping surface features $\boldsymbol{p_2}$ to ground level $\boldsymbol{p_1}$ in the image plane.

$$\dot{\boldsymbol{p}} = \boldsymbol{m} \begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix} = s\boldsymbol{h} \begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix} \tag{7}$$

This representation for geocentric pose is valid for satellite images with locally affine imaging geometry; however, often satellite images are distributed in orthorectified form. Orthorectification for a single image is a pixel remapping that approximates orthographic, z-axis aligned, projection with fixed pixel scale (meters) and minimizes terrain relief displacement using a ground-level elevation model [28]. For an affine camera, we define orthorectification as the pixel remapping w_Z acquired by inverting 2x3 affine matrix $\boldsymbol{A}$ with elevation model $\boldsymbol{z_0}$ and scaling by $\boldsymbol{K} = diag(k, k)$ defined by the desired pixel scale.

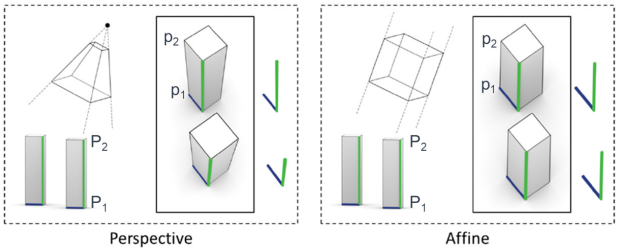

Figure 3: Affine projection $\boldsymbol{A}$ maps world coordinates $\boldsymbol{P}$ to image coordinates $\boldsymbol{p}$, preserving invariant properties of parallelism and ratio of lengths on parallel lines.

$$w_Z\left(\boldsymbol{p}; \boldsymbol{z_0}\right) = \boldsymbol{K} \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}^{-1} \left[\boldsymbol{p} - \begin{pmatrix} a_{13} \\ a_{23} \end{pmatrix} \boldsymbol{z_0} \right] \tag{8}$$

This function is invertible given the elevation model and camera metadata. For scenes with little terrain relief such that $\boldsymbol{z_0}$ is approximately a constant or linear function, the orthorectified projection is approximately affine with equivalent scale and angle values for estimating geocentric pose. We demonstrate only a modest reduction in accuracy for our model applied to orthorectified images (Tab. 8).

4. Methods

4.1. Model and Optimization

We estimate geocentric pose $g(\boldsymbol{I}) = \{s, \theta, \boldsymbol{h}\}$ with scale prediction defined by the known relationship between magnitude and height defined in (6). We parameterize our model in a multi-task deep network with a ResNet-34 encoder [17] and U-Net decoder [34] and supervise training with known height AGL values derived from lidar and multi-view stereo (MVS). We attach independent output heads for height and vector field magnitudes to the decoder, with consistency encouraged by solving for image-level scale as described below and shown in Fig. 4. This overcomes one of the most significant weaknesses of [6] which relates height as a prior for magnitude with an insufficient mechanism to learn the known relationship.

We represent vector fields with magnitude $\boldsymbol{m}$ and angle θ encoded as the vector $(\cos\theta, \sin\theta)$ to avoid ambiguity at zero. We explicitly supervise learning of both magnitude $\boldsymbol{m}$ and height $\boldsymbol{h}$ as orthogonal observations to enrich the features embedded in the encoder and decoder layers. We also explicitly supervise learning of scale which provides a gradient for learning to predict consistent magnitude and height values from images with or without explicit height labels. For some applications, scale will be avail-

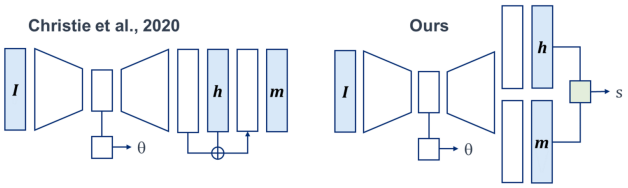

Figure 4: Christie et al. [6] relate height as a prior for vector magnitude. Our model explicitly encodes the known affine relationship with a custom least squares solver layer.

able in metadata, so in addition to helping enforce consistency during training, having it as a prediction at test time can help identify regions where the model is performing poorly. Scale is predicted in the model with a custom least squares solver layer implemented using the pseudo-inverse, $s = \left(\boldsymbol{h}^T\boldsymbol{h}\right)^{-1}\boldsymbol{h}^T\boldsymbol{m}$.

The total loss L minimized in training is a weighted sum of mean squared error (MSE) losses for all terms, $L = f_\theta L_\theta + f_s L_s + f_{\boldsymbol{h}} L_{\boldsymbol{h}} + f_{\boldsymbol{m}} L_{\boldsymbol{m}}$. For height and magnitude, MSE is implemented as the mean of MSE values for each labeled image in a batch to both reduce sensitivity to unlabeled pixels and allow for training images without height labels to directly supervise height and magnitude; in the latter case, height and magnitude MSE losses are zero and only the scale loss is back-propagated through those layers. We set weighting factors f_θ=10, f_s=10, $f_{\boldsymbol{h}}$=1, and $f_{\boldsymbol{m}}$=2 to normalize value ranges. To improve training efficiency and allow for a larger batch size, we down-sample input images by a factor of two. Our batch size b=8 speeds convergence for angle prediction since each training image provides only a single target sample. We use the Adam optimizer [18] with a learning rate of 1e-4 which we found to improve convergence. For all experiments, we trained models for 200 epochs.

4.2. Augmentation

The distributions for angle of parallel projection θ, scale factor s relating height and magnitude, and object height $\boldsymbol{h}$ are all heavily biased. Angle and scale are biased by the limited viewing geometries from satellite orbits, and very tall objects are rare. We encourage generalization and address bias with image remap augmentations w_θ, w_s, and w_h.

$$w_\theta\left(\boldsymbol{p};\dot{\theta}\right) = \begin{pmatrix} \cos\dot{\theta} & \sin\dot{\theta} \\ -\sin\dot{\theta} & \cos\dot{\theta} \end{pmatrix}\boldsymbol{p} \tag{9}$$

$$w_s\left(\boldsymbol{p};\dot{s}\right) = \begin{pmatrix} \dot{s} & 0 \\ 0 & \dot{s} \end{pmatrix}\boldsymbol{p} \tag{10}$$

$$w_h\left(\boldsymbol{p};\dot{h}\right) = \boldsymbol{p} + \dot{\boldsymbol{m}}\begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix} \tag{11}$$

$$\dot{\boldsymbol{m}} = -\,s\left(\boldsymbol{h} + \dot{h}\right) \tag{12}$$

While image augmentations for rotation w_θ and scale w_s are commonly applied to regularize training in deep networks and depend only on image coordinates, our height augmentation w_h inverts geocentric pose vectors to synthetically increase building heights (Fig. 5). Note that shadows are not adjusted by this simple but effective augmentation. This does not appear to be an impediment, and we believe that over-reliance on shadows should not be encouraged in learning because they are very often not observed.

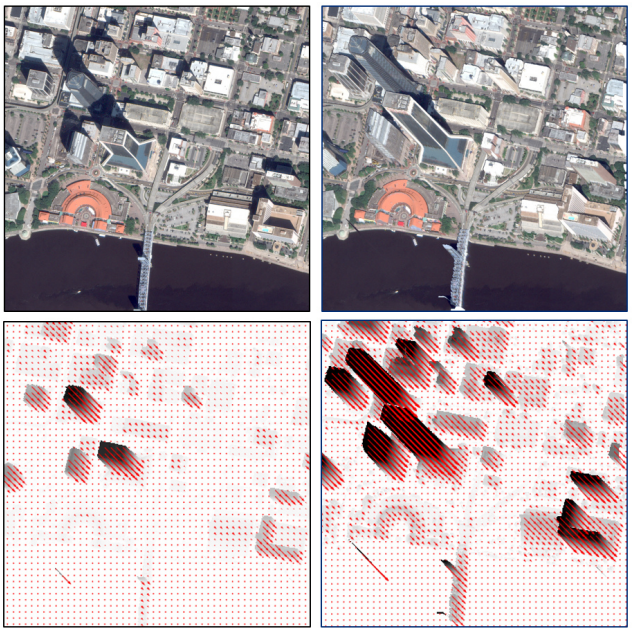

Figure 5: Affine geocentric pose enables height augmentations to address the long-tailed nature of height distributions in the data. For this example, we remap above-ground features in the original image (left) with 2.3x height (right) for RGB images (top) and geocentric pose values (bottom).

4.3. MVS Supervision

The geocentric pose method presented in [6] relies entirely on normalized digital surface model (nDSM) products derived from airborne lidar to define height above ground for supervising learning; however, lidar is not available or practical to collect in many world regions. MVS methods for satellite images [10, 41, 37, 30, 33, 35] offer a more widely available alternative to lidar for training our model, and accuracy for these methods is comparable for larger features relevant to mapping applications [3, 22, 20]. To explore this, we adapted the Urban Semantic 3D (US3D) rendering pipeline [2] to produce labeled datasets using height derived from MVS. In our experiments, we demonstrate that models trained with lidar and commercially acquired MVS derived heights both perform well (Tab. 9).

5. Experiments

5.1. Datasets

We demonstrate our methods with the US3D public dataset [2], first developed for the 2019 IEEE GRSS Data Fusion Contest (DFC19) [21] and later extended by Christie et al. [6] to explore the geocentric pose regression task. US3D includes satellite images and lidar-derived reference labels covering Jacksonville, Florida (JAX), Omaha, Nebraska (OMA), and Atlanta, Georgia (ATL). US3D images are 2048x2048 pixels except for those over ATL with off-nadir angle greater than 35 degrees. Those images were cropped much smaller, making them inconsistent with the rest of the dataset. Far off-nadir commercial satellite images are uncommon and often undesirable, as described by [40]. For better consistency, we excluded the cropped images for training and testing our models.

We further extended the public US3D dataset to include satellite images of San Fernando, Argentina (ARG) from the 2016 Multi-View Stereo 3D Mapping Challenge [3]. Until now, US3D was limited to cities within the United States with Western architecture. ARG presents additional challenges, with fewer tall buildings and increased diversity in architectural styles, including very closely spaced buildings. Our current model does not perform as well for ARG as for the other sites, and we hope that publicly releasing this data will encourage further research to improve performance. Our extended dataset is illustrated in Fig. 6. Data statistics are provided in a supplement [1].

5.2. Results

State of the Art for Geocentric Pose: We first demonstrate that our model dramatically improves upon the state of the art method presented by Christie et al. [6] which, to the best of our knowledge, is the only published work on this task. Their FLOW-H model was trained without augmentations and FLOW-HA with rotation augmentations. We present results for our model trained with and without augmentations and trained with only the DFC19 train set including Jacksonville and Omaha, with only the ATL train set, and with the train sets from all four cities. Results are shown in Tab. 1 and Fig. 7 for the DFC19 test set and in Tab. 2 for the ATL test set. We report root mean square error (RMSE) values for magnitude (pixels), angle (degrees), height (meters), and endpoint error (pixels). Mean absolute error (MAE) was reported by [6], so we also demonstrate our improvements in terms of MAE in a supplement [1]. Observe that our model design and augmentations each improve performance and reduce over-fitting for DFC19 but that augmentations sometimes harm performance for ATL. We believe this is because both train and test ATL images were drawn from a much less diverse single pass, same day satellite collection over a single city. Also note that training on additional cities consistently improves angle predictions for both test sets.

Method	Train	Mag	Angle	Endpoint	Height
FLOW-HA [6]	DFC19	7.08	29.05	7.57	6.12
FLOW-H [6]	DFC19	6.12	21.84	7.09	5.61
Ours-NoAug	DFC19	5.06	16.53	5.58	4.81
Ours	DFC19	**4.07**	13.73	4.31	4.00
Ours-NoAug	All Cities	5.19	20.51	5.91	4.86
Ours	All Cities	4.14	**11.79**	**4.29**	**3.89**

Table 1: Our method improves on state of the art RMSE for the DFC19 test set.

Method	Train	Mag	Angle	Endpoint	Height
FLOW-HA [6]	ATL	7.53	20.54	8.19	10.58
FLOW-H [6]	ATL	5.62	14.41	5.88	8.95
Ours-NoAug	ATL	3.50	11.00	3.93	4.88
Ours	ATL	**3.45**	13.19	3.88	4.89
Ours-NoAug	All Cities	3.46	**10.67**	3.83	**4.84**
Ours	All Cities	3.50	**10.67**	**3.72**	4.95

Table 2: Our method improves on state of the art RMSE for the ATL test set.

Method	Mag	Angle	Endpoint	Height
FLOW-HA [6]	3.82	31.96	4.21	3.09
Ours-NoAug	3.97	**19.89**	4.14	3.52
Ours	**3.32**	23.28	**3.56**	**3.00**

Table 3: Our method improves on state of the art RMSE for the challenging new ARG test site. All models were trained on all four sites.

For both test sets, our model outperforms FLOW-HA even with no augmentations. We believe a primary contributing factor to [6] underestimating heights for taller buildings is its inadequate modeling of the relationship between height and vector field magnitude. As evidenced by our metric results and visually in Fig. 7, our model that more directly relates height and magnitude with an image-level scale factor produces more accurate predictions.

To evaluate FLOW-HA performance for our new ARG test set, we trained it from scratch with all four cities and compared to our model both with and without augmentations (Tab. 3). Since very few buildings in this region are tall, FLOW-HA is more competitive with our model trained without augmentations; however, our model trained with augmentations provides a significant improvement.

State of the Art for Height Prediction: We also compare our results with methods for pixel-level height prediction using both our full geocentric pose model and the same model with only a height prediction head. For fair compari-

Figure 6: US3D includes satellite images from WorldView 2 and WorldView 3 and covers a variety of geographic location, season, viewpoint, and resolution. We added data for San Fernando, Argentina which presents new challenges, with fewer tall buildings and increased architectural diversity. Example images are shown. Statistics are provided in a supplement [1].

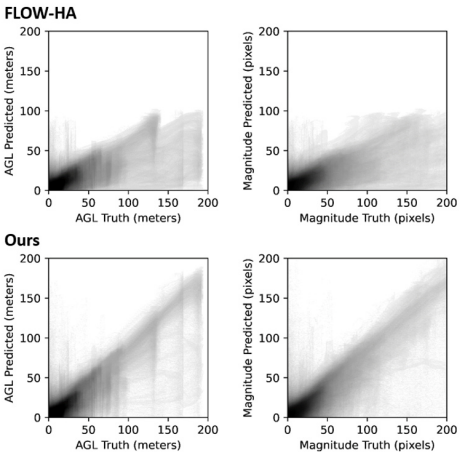

Figure 7: AGL heights and vector magnitudes are compared to reference values for the DFC19 test set. Pixel intensity indicates count. Our model significantly improves upon FLOW-HA [6], particularly for infrequent tall buildings.

Method	MAE	RMSE
Srivastava et al. [38]	3.74	5.85
Mou and Zhu [29]	3.62	5.40
Mahmud et al. [27]	3.34	5.02
Christie et al. [6]	2.76	4.33
Ours (Height)	**2.52**	**3.89**
Ours (Full)	2.78	4.07

Table 4: Our pixel-level building height predictions improve on state of the art MAE and RMSE (meters) for the custom DFC19 test set used by [27].

Method	All MAE	All RMSE	Bldgs MAE	Bldgs RMSE
Kunwar [19]	2.69	9.26	8.33	19.65
Zheng et al. [43]	2.94	9.24	8.72	19.32
Christie et al. [6]	2.98	8.23	7.73	16.87
Ours (Height)	**2.31**	**5.46**	**5.75**	**10.69**
Ours (Full)	2.44	5.76	6.14	11.35

Table 5: Our pixel-level height predictions improve on state of the art MAE and RMSE (meters) for the DFC19 test set used for [20].

son with [27, 29, 38, 6], we retrained and tested our models and [6] using the custom DFC19 train and test sets used in [27]. For comparison with [19, 43, 6], we tested against the DFC19 test set used for [20]. Results are reported in Tab. 4 and Tab. 5, respectively. Both models achieve state of the art accuracy, with our single-task model outperforming the multi-task model. We attribute this improvement to our augmentations that address biases in the data.

Geographic Diversity: We assess model performance for geographically diverse cities shown in Fig. 6. Results are shown in Tab. 6 for our full model, the same model trained without augmentations, a version with only a height regression head, and our model fine tuned for twenty epochs on each individual city. We report RMSE for angle, scale, height, and vector field endpoints. Here we see a modest improvement in height prediction with our full model, though not for ATL which includes only images from a single pass as discussed above.

We might expect fine tuning our model for each site to notably improve performance given unique geographic appearance and differences in the image resolutions among sites. After fine tuning, we see similar or modestly improved performance for all sites except for OMA which has limited appearance diversity.

In a supplement [1], we also report the R^2 metric for height and endpoint and discuss its value for comparing performance among cities. Height prediction is especially challenging for ARG due to both closely spaced residential

Prediction	City	Full	NoAug	Height	Tuned
Angle (deg)	JAX	12.48	20.40	N/A	**10.26**
	OMA	**11.31**	20.58	N/A	17.90
	ATL	10.67	10.67	N/A	**9.89**
	ARG	23.28	**19.89**	N/A	20.48
Scale (pix/m)	JAX	**0.11**	0.17	N/A	**0.11**
	OMA	**0.11**	0.16	N/A	**0.11**
	ATL	0.10	0.12	N/A	**0.08**
	ARG	0.16	0.19	N/A	**0.13**
Height (m)	JAX	3.33	3.69	3.67	**3.23**
	OMA	**4.15**	5.51	4.20	4.32
	ATL	4.86	4.84	4.73	**4.67**
	ARG	**3.00**	3.52	3.27	3.06
Endpoint (pix)	JAX	3.61	3.69	N/A	**3.53**
	OMA	**4.63**	6.80	N/A	5.19
	ATL	3.66	3.83	N/A	**3.45**
	ARG	**3.56**	4.14	N/A	3.58

Table 6: RMSE is compared with our full model, without augmentations, with only a height prediction head, and with the full model fine tuned for each city.

Prediction	Test Aug	Ours	NoAug
Angle (deg)	None	17.73	**17.72**
	Scale	**19.00**	34.93
	Angle	**20.18**	87.72
	Height	**14.61**	17.92
Scale (pix/m)	None	**0.13**	0.17
	Scale	**0.18**	0.28
	Angle	**0.15**	0.29
	Height	**0.17**	**0.17**
Height (m)	None	**3.84**	4.67
	Scale	**4.24**	5.34
	Angle	**4.20**	8.17
	Height	**6.18**	7.27
Endpoint (pix)	None	**3.84**	4.84
	Scale	**5.05**	7.17
	Angle	**4.17**	9.76
	Height	**5.47**	7.07

Table 7: Test-time augmentations demonstrate improved RMSE over a range of conditions not observed in the test set. Results are for test images from all four cities.

housing and the infrequency and relative uniqueness of the taller buildings. While RMSE is low relative to other sites due to the predominance of low height buildings, the R^2 metric, which is normalized by the value range, is poor due to inaccurate predictions for rare tall structures.

Test-time Augmentations: To better understand our model's ability to generalize, we tested models trained with and without augmentations on all cities with random rotation, scale, and height augmentations. Results in Tab. 7 show significant improvement in generalization to account for a broader range of conditions than observed in the limited test sets. Our height augmentations help overcome the issue of long-tailed height distributions in the training data by significantly improving height estimation both with and without test-time height augmentations; however, it does not learn to generalize for these appearance changes as well as for scale and rotation angle. Also note that angle predictions are more accurate for images with test-time height augmentations because angles are more observable when vector magnitudes are larger.

Orthorectified Images: As discussed in Sec. 3, satellite images are often orthorectified for convenient dissemination and georeferencing on a Cartesian grid. This orthorectification distorts images such that our affine assumptions are violated. To evaluate the impact on model performance, we applied orthorectification to all test images and compared accuracies to the original test images as shown in Tab. 8. Angle prediction is improved due to a smaller value range, but scale accuracy is degraded significantly. We then orthorectified the training images and fine tuned the model for twenty epochs. The resulting predictions are only modestly less accurate than their affine counterparts, indicating that our model can produce accurate predictions when applied to typical orthorectified images. Rare high terrain slopes (meters) are not present in our data set but can induce more significant rectification errors (pixels) in orthorectified images. We believe modest changes to our model implementation can address this issue, and we plan to pursue this in future work.

Building Segmentation: We compare performance with non-building above-ground feature heights set to zero in train and test sets (Tab. 8). Performance is comparable for this narrowly-defined task, suggesting that our method can be used for building segmentation and rectification when semantic labels are available for training.

Rectification to Ground Level: Rectifying building segmentation labels into footprints is one of the applications of our work. We demonstrate the ability of our model to perform this task by plotting instance-level intersection over union (IoU) as a function of maximum vector magnitude inside each instance for both unrectified building labels and our rectified outputs compared to building footprints. We also show RMS IoU as a function of a maximum magnitude threshold for building instances included in the calculation (Fig. 8). In both cases, we evaluate a subset of the building test instances where warping with reference geocentric pose values achieves a minimum IoU of 0.9. This helps eliminate instances where occlusion prevents accurate geocentric pose regression.

Test Set	Tuned	Angle	Scale	Height	Endpoint
All pixels	No	17.73	**0.13**	**3.84**	**3.84**
All pixels ortho	No	12.99	0.26	4.32	5.41
All pixels ortho	Yes	**10.65**	0.19	3.95	4.57
Building pixels	No	17.81	0.14	**3.48**	**3.41**
Buildings ortho	No	**10.00**	0.19	4.78	5.42
Buildings ortho	Yes	10.70	**0.13**	3.94	4.17

Table 8: Predictions for orthorectified images where affine assumptions are violated are only modestly less accurate than their affine counterparts after fine tuning. Metrics shown are RMSE. All pixels results are for all four cities. Results for only building pixels do not include ARG.

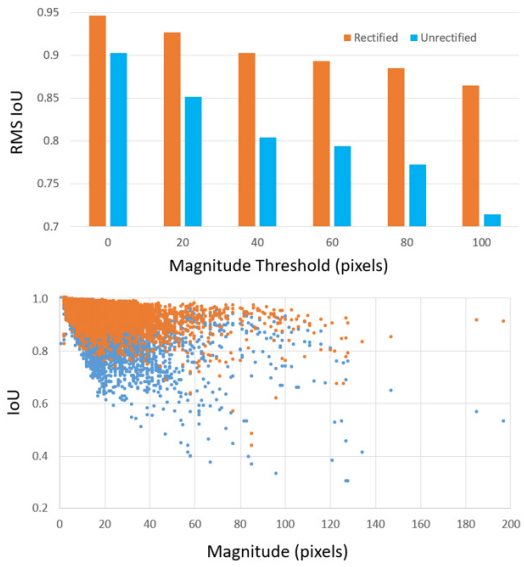

Figure 8: Instance-level building IoU improvement with rectification is shown versus max vector magnitude inside the instance (bottom), and RMS IoU is shown for instances with max magnitude above a threshold (top).

MVS Supervision: For experiments to supervise learning with MVS instead of lidar, discussed in Sec. 4.3, we used commercial Vricon MVS products that overlap the DFC19 Jacksonville and Omaha train and test sets. While our current MVS dataset is limited to a subset of the DFC19 images (1461/225 train/test images), MVS data can be produced over large scales anywhere in the world from satellite images. We believe this is important for providing sufficient diversity in training to promote generalization.

We trained our model separately with lidar and MVS and then tested against both lidar and MVS reference values. As shown in Tab. 9, both models achieved comparable accuracies for both heights and vector field endpoints across both evaluation types. Models trained using one source of reference data performed better when evaluating against the same source (e.g., train and test with lidar), but both models performed well. Angle and scale predictions appear surprisingly less accurate for the model trained with MVS; however, these are image-level predictions, so the differences for only 225 test images are unlikely to be significant.

Train	Test	Angle	Scale	Height	Endpoint
Lidar	Lidar	**13.72**	**0.13**	4.56	5.01
Lidar	MVS	**13.72**	**0.13**	5.49	5.96
MVS	Lidar	19.23	0.15	5.06	5.64
MVS	MVS	19.23	0.15	**4.52**	**4.94**

Table 9: RMSE for models supervised with MVS are comparable to those supervised with lidar.

6. Discussion

Geocentric pose regression from oblique monocular remote sensing images has the potential to dramatically improve the utility of oblique images for mapping applications. In this work, we presented our method that exploits invariant properties of affine imaging geometry to achieve state of the art performance for this task and also for pixel-level height prediction. Our affine approximation is very accurate for the 2048 x 2048 pixel sub-images we extract from large satellite images. For efficient processing of full images, our method can be applied in parallel to overlapping image tiles.

We also addressed challenges that must be overcome to deploy a solution in the real world, including supervising learning without lidar, reducing bias in training to enable generalization, and ensuring good performance for orthorectified images which are commonly available but which violate affine assumptions. Results rectifying images, building labels, and heights to ground level indicate the value of our approach for a range of mapping tasks.

Our open source code provides a strong baseline for public leaderboard evaluation. In a supplement [1], we provide additional details of the public dataset [11] and metrics for evaluation as well as examples of common failure cases to motive further research.

Acknowledgements

This work was supported by the National Geospatial-Intelligence Agency and approved for public release, 21-483, with distribution statement A – approved for public release; distribution is unlimited. Commercial satellite images were provided courtesy of DigitalGlobe.

References

[1] *Monocular Geocentric Pose, GitHub repository*, 2020. https://github.com/pubgeo/monocular-geocentric-pose. 5, 6, 8

[2] Marc Bosch, Kevin Foster, Gordon Christie, Sean Wang, Gregory D Hager, and Myron Brown. Semantic Stereo for Incidental Satellite Images. In *WACV*, 2019. 4, 5

[3] M. Bosch, Z. Kurtz, S. Hagstrom, and M. Brown. A multiple view stereo benchmark for satellite imagery. In *Applied Imagery Pattern Recognition (AIPR) Workshop*, 2016. 4, 5

[4] Honglie Chen, Weidi Xie, Andrea Vedaldi, and Andrew Zisserman. AutoCorrect: Deep Inductive Alignment of Noisy Geometric Annotations. *BMVC*, 2019. 1

[5] Yanhua Cheng, Rui Cai, Zhiwei Li, Xin Zhao, and Kaiqi Huang. Locality-Sensitive Deconvolution Networks with Gated Fusionfor RGB-D Indoor Semantic Segmentation. In *CVPR*, 2017. 2

[6] Gordon Christie, Rodrigo Rene Rai Munoz Abujder, Kevin Foster, Shea Hagstrom, Gregory D Hager, and Myron Z Brown. Learning Geocentric Object Pose in Oblique Monocular Images. In *CVPR*, 2020. 1, 2, 3, 4, 5, 6

[7] Carlo de Franchis, Enric Meinhardt-Llopis, Julien Michel, J-M Morel, and Gabriele Facciolo. On stereo-rectification of pushbroom images. In *ICIP*, 2014. 2

[8] Ilke Demir, Krzysztof Koperski, David Lindenbaum, Guan Pang, Jing Huang, Saikat Basu, Forest Hughes, Devis Tuia, and Ramesh Raska. DeepGlobe 2018: A Challenge to Parse the Earth through Satellite Images. In *CVPRW*, 2018. 1

[9] Jigar Doshi, Saikat Basu, and Guan Pang. From Satellite Imagery to Disaster Insights. *NeurIPS Workshops*, 2018. 1

[10] G. Facciolo, C. De Franchis, and E. Meinhardt-Llopis. Automatic 3d reconstruction from multi-date satellite images. In *CVPRW*, 2017. 4

[11] Kevin Foster, Gordon Christie, and Myron Brown. *IEEE Dataport: Urban Semantic 3D Dataset*, 2020. http://dx.doi.org/10.21227/9frn-7208. 8

[12] Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, and Dacheng Tao. Deep ordinal regression network for monocular depth estimation. In *CVPR*, pages 2002–2011, 2018. 2

[13] Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J Brostow. Digging into self-supervised monocular depth estimation. In *ICCV*, 2019. 2

[14] Saurabh Gupta, Pablo Arbelaez, and Jitendra Malik. Perceptual organization and recognition of indoor scenes from RGB-D images. In *CVPR*, 2013. 1, 2

[15] Saurabh Gupta, Ross Girshick, Pablo Arbelaez, and Jitendra Malik. Learning rich features from RGB-D images for object detection and segmentation. In *ECCV*, 2014. 2

[16] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2003. 3

[17] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 3

[18] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2015. 4

[19] Saket Kunwar. U-Net Ensemble for Semantic and Height Estimation Using Coarse-Map Initialization. In *IGARSS*, 2019. 2, 6

[20] S. Kunwar, H. Chen, M. Lin, H. Zhang, P. Dangelo, D. Cerra, S. M. Azimi, M. Brown, G. Hager, N. Yokoya, R. Hansch, and B. Le Saux. Large-scale semantic 3d reconstruction: Outcome of the 2019 ieee grss data fusion contest - part a. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2020. 2, 4, 6

[21] Bertrand Le Saux, Naoto Yokoya, Ronny Hansch, Myron Brown, and Greg Hager. 2019 Data Fusion Contest [technical committees]. *IEEE Geoscience and Remote Sensing Magazine*, 2019. 5

[22] Matthew J. Leotta, Chengjiang Long, Bastien Jacquet, Matthieu Zins, Dan Lipsa, Jie Shan, Bo Xu, Zhixin Li, Xu Zhang, Shih-Fu Chang, Matthew Purri, Jia Xue, and Kristin Dana. Urban semantic 3d reconstruction from multiview satellite imagery. In *CVPRW*, June 2019. 4

[23] Zhengqi Li and Noah Snavely. MegaDepth: Learning Single-View Depth Prediction from Internet Photos. In *CVPR*, 2018. 2

[24] Di Lin, Guangyong Chen, Daniel Cohen-Or, Pheng-Ann Heng, and Hui Huang. Cascaded Feature Network for Semantic Segmentation of RGB-D Images. In *ICCV*, 2017. 2

[25] Shice Liu, Yu Hu, Yiming Zeng, Qiankun Tang, Beibei Jin, Yinhe Han, and Xiaowei Li. See and Think: Disentangling Semantic Scene Completion. In *NeurIPS*, 2018. 2

[26] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully Convolutional Networks for Semantic Segmentation. In *CVPR*, 2015. 2

[27] Jisan Mahmud, True Price, Akash Bapat, and Jan-Michael Frahm. Boundary-Aware 3D Building Reconstruction From a Single Overhead Image. In *CVPR*, 2020. 1, 2, 6

[28] J.C. McGlone, E.M. Mikhail, J.S. Bethel, R. Mullen, American Society for Photogrammetry, and Remote Sensing. *Manual of Photogrammetry*. American Society for Photogrammetry and Remote Sensing, 2004. 3

[29] Lichao Mou and Xiao Xiang Zhu. IM2HEIGHT: Height Estimation from Single Monocular Imagery via Fully Residual Convolutional-Deconvolutional Network. *arXiv:1802.10249*, 2018. 1, 2, 6

[30] Ö. C. Özcanli, Y. Dong, J. Mundy, Helen F. Webb, R. Hammoud, and V. Tom. A comparison of stereo and multiview 3-d reconstruction using cross-sensor satellite imagery. *CVPRW*, pages 17–25, 2015. 4

[31] Seong-Jin Park, Ki-Sang Hong, and Seungyong Lee. RDFNet: RGB-D Multi-level Residual Feature Fusion forIndoor Semantic Segmentation. In *ICCV*, 2017. 2

[32] Xiaojuan Qi, Renjie Liao, Jiaya Jia, Sanja Fidler, and Raquel Urtasun. 3D Graph Neural Networks for RGBD Semantic Segmentation. In *ICCV*, 2017. 2

[33] R. Qin. Rpc stereo processor (rsp) – a software package for digital surface model and orthophoto generation from satellite stereo imagery. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, pages 77–82, 2016. 4

[34] O. Ronneberger, P.Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, volume 9351 of *LNCS*, pages 234–241. Springer, 2015. (available on arXiv:1505.04597 [cs.CV]). 3

[35] E. Rupnik, M. Pierrot-Deseilligny, and Alexandre Delorme. 3d reconstruction from multi-view vhr-satellite images in micmac. *ISPRS Journal of Photogrammetry and Remote Sensing*, 139:201–211, 2018. 4

[36] Max Schwarz, Anton Milan, Arul Selvam Periyasamy, and Sven Behnke. RGB-D Object Detection and Semantic Segmentation for Autonomous Manipulation in Clutter. *International Journal of Robotics Research*, 2018. 2

[37] D. Shean, O. Alexandrov, Z. Moratto, B. Smith, I. Joughin, C. Porter, and P. Morin. An automated, open-source pipeline for mass production of digital elevation models (dems) from very-high-resolution commercial stereo satellite imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 116:101–117, 2016. 4

[38] Shivangi Srivastava, Michele Volpi, and Devis Tuia. Joint height estimation and semantic labeling of monocular aerial images with cnns. In *IGARSS*, 2017. 1, 2, 6

[39] Weiyue Wang and Ulrich Neumann. Depth-aware CNN for RGB-D Segmentation. In *ECCV*, 2018. 2

[40] Nicholas Weir, David Lindenbaum, Alexei Bastidas, Adam Van Etten, Sean McPherson, Jacob Shermeyer, Varun Kumar, and Hanlin Tang. SpaceNet MVOI: a Multi-View Overhead Imagery Dataset. In *ICCV*, 2019. 1, 5

[41] Kai Zhang, Jin Sun, and Noah Snavely. Leveraging Vision Reconstruction Pipelines for Satellite Imagery. In *ICCVW*, 2019. 4

[42] Shanshan Zhao, Huan Fu, Mingming Gong, and Dacheng Tao. Geometry-aware symmetric domain adaptation for monocular depth estimation. In *CVPR*, 2019. 2

[43] Zhuo Zheng, Yanfei Zhong, and Junjue Wang. Pop-Net: Encoder-Dual Decoder for Semantic Segmentation and Single-View Height Estimation. In *IGARSS*, 2019. 2, 6