STELLAR: A LARGE SATELLITE STEREO DATASET FOR DIGITAL SURFACE MODEL GENERATION

Sonali Patil[1,*] and Qi Guo[2]

[1] German Aerospace Center (DLR), Braunschweig, Germany - sonali.patil@dlr.de
[2] Purdue University, School of Electrical and Computer Engineering, West Lafayette, IN, USA - guo675@purdue.edu

KEY WORDS: Photogrammetry, digital surface model, big data challenge, digital earth, computationally intensive data processing.

ABSTRACT:

Stellar is a large, satellite stereo dataset. It contains rectified stereo pairs of the terrain captured by the satellite image sensors and corresponding true disparity maps and semantic segmentation. Unlike stereo vision in autonomous driving and mobile imaging, a satellite stereo pair is not captured simultaneously. Thus, the same object in a satellite stereo pair is more likely to have a varied visual appearance. Stellar provides flexible access to such stereo pairs to train methods to be robust to such appearance variation. We use publicly available data sources, and invented several techniques to perform data registration, rectification, and semantic segmentation on the data to build Stellar. In our preliminary experiment, we fine-tuned two deep-learning stereo methods on Stellar. The result demonstrates that most of the time, these methods generate denser and more accurate disparity maps for satellite stereo by fine-tuning on Stellar, compared to without fine-tuning on satellite stereo datasets, or fine-tuning on previous, smaller satellite stereo datasets. Stellar is available to download at `https://github.com/guo-research-group/Stellar`.

1. INTRODUCTION

Digital Surface Models (DSMs) contain 3D representations of the earth’s surface. It records the shape information of terrains and human constructions of an area at specific time stamps. DSMs have broad applications in environmental studies, for example, monitoring and analyzing glaciers melting, water level changes of rivers and lakes, and human activities. Additionally, they are widely used in non-environmental applications such as urban visualization, infrastructure planning, and mixed reality.

There are two major approaches to generating DSMs nowadays. The first approach uses Lidars on UAVs to scan the landscape to produce point clouds, which are then processed to become DSMs. The DSMs from this approach have an average height error as low as 0.3 centimeters (San Diego LiDAR Report, n.d.) thanks to the high accuracy of Lidar sensors. But because the cruising altitude of UAVs is relatively low, the Lidar scanning can only cover a limited area each time. A typical full swath spacing of a UAV Lidar scan is below 1km. Thus, it is slow and expensive to build large-area DSMs via this approach. Furthermore, Lidar scanning relies on GPS for accurate geolocation, but the reliability of GPS signals can be easily affected by landscapes, atmospheric conditions, etc., in practice.

Thanks to the recent maturation of very high resolution (VHR) satellite image sensors, an alternative approach that reconstructs DSMs via stereo matching emerges. In this approach, people first perform two-view or multi-view stereo matching using VHR images of the same area captured from different satellite perspectives to generate local disparity maps. Then they fuse disparity maps of areas together to produce a DSM of a larger area. A VHR image can cover areas of hundreds of sq. km with a ground sampling resolution between 30cm-60cm depending on the viewing angle. Therefore, despite its lower accuracy compared to Lidar scanning, it is cheaper and more efficient for people to create a global DSM via satellite stereo.

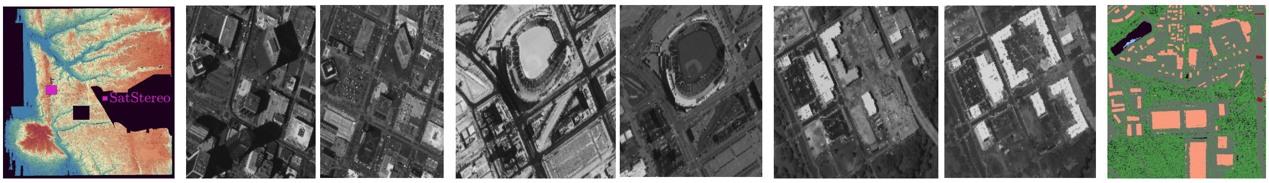
The accuracy of stereo matching algorithms has improved these years significantly in street view and indoor settings thanks to the appearance of large, supervised datasets such as KITTI (Menze and Geiger, 2015) and Middlebury (Scharstein et al., 2014). These datasets provide sufficient stereo image pairs and corresponding ground truth disparity maps to train data-hungry deep-learning (DL) methods. However, in the domain of satellite stereo, developing a large-scale supervised dataset with diverse VHR images that supports quantitative evaluation is still at an initial stage in our perspective.

We present *Stellar*. It is a large satellite stereo dataset that contains stereo VHR image pairs and corresponding ground truth disparity maps generated from Lidar measurements. Stellar is so far the largest in areas of data to the best of our knowledge (Table 1). It is flexible to use. People can select stereo VHR image pairs that have specific time differences or sun angle differences. This allows users to control the appearance difference between the stereo image pairs caused by seasonal vegetation, weather, human activities, and shadows and specular reflections caused by sunlight (Figure 1). Stellar also provides semantic segmentation of the areas (*e.g.*, Figure 1), which enables quantitative analysis of stereo-matching accuracy based on types of areas, e.g., buildings, rivers, and vegetation.

Although the source data of Stellar comes from several public repositories (Bosch et al., 2016, Brown et al., 2018, Van Etten et al., 2018), there are image processing challenges that need to be overcome to build Stellar. First, the Lidar measurements and VHR images use different coordinate reference systems. This means the elevation measured from Lidar has to be calibrated to the same coordinate system as VHR images. We used a novel co-registration method to perform the alignment (Section 3.) Second, it is non-trivial to rectify stereo pairs for VHR images, as VHR sensors on satellites have nonlinear homography and non-conjugate epipolar curves. In Stellar, we developed a patch-based rectification method to overcome this challenge (Section 3.2.3.) Finally, Lidar measurements and VHR images

[*] Corresponding author. This work was done at Purdue University.

This contribution has been peer-reviewed.
https://doi.org/10.5194/isprs-archives-XLVIII-M-1-2023-433-2023 | © Author(s) 2023. CC BY 4.0 License.



(a) Stellar Sample DSM (b) Sun angle difference. (c) Seasonal changes. (d) Human activities. (e) Semantic segmentation.

Figure 1. From left to right, (a) Stellar sample DSM as compared to a SatStereo region, (b-d) stereo pairs with varied appearances due to sun angle, season, and human activities. (e) Typical semantic segmentation. Orange for buildings, red for bridges, blue for water, green for vegetation and gray for ground.

Dataset	Area (km^2)	#City	Best Res. (m)	Unipolar Disparity	Pair Selection	Semantic Labels	Varying Stereo Image Size	Lidar DSM	True Disparity	Multi Date	Stereo Baseline Variation
MVS3DM (Bosch et al., 2016)	1.5	1	0.3	✗	✗	✗	✗	✓	✗	✓	✗
DFC (Bosch et al., 2019)	7.15	2	0.3	✗	✗	✓	✗	✓	✓	✓	✓
SatStereo (Patil et al., 2019a)	4.5	3	0.3	✗	✗	✓	✓	✗	✓	✓	✓
WHU-Stereo (Li et al., 2022b)	≈ 778	**6**	0.65	✗	✗	✗	✗	✗	✓	✗	✗
Stellar (ours)	**1,682**[1]	5	0.3	✓	✓	✓	✓	✓	✓	✓	✓

Table 1. Stellar vs. existing satellite stereo datasets. Apart from covering much larger areas than previous ones, Stellar allows users to select stereo image pairs with specific time differences and sun angle differences, etc. Stellar also supports generating stereo images with variable sizes and provides dense ground truth disparity.

of the same area may be acquired with a time difference longer than a year. The landscape may vary between the Lidar and the VHR image, and among VHR images (Figure 1). This creates noise in the ground truth disparity map, as the disparity from Lidar may not reflect the true disparity of a stereo VHR image pair. To mitigate this issue, Stellar provides semantic segmentation along with the ground truth disparity maps, which enables users to analyze methods on areas less prone to change over time (Section refsubsubsec:semantic.)

To validate the effectiveness of Stellar, we fine-tune two DL stereo matching algorithms on it and on previous satellite stereo datasets, and analyze their performance. Our experiments demonstrate that, in most cases, the DL stereo methods improve their accuracy on satellite stereo by fine-tuning on Stellar, compared to without fine-tuning or fine-tuning on previous, smaller satellite stereo datasets. The detailed description of our experiments is in Section 4.

Stellar is available to download at `https://github.com/guo-research-group/Stellar`.

2. RELATED WORK

2.1 Deep Learning (DL) Satellite Stereo

The classic two-stage stereo matching pipeline first computes a cost volume from a pair of rectified images, then estimates the disparity map based on the cost volume. Traditionally, semi-global matching (SGM) (Hirschmuller, 2005) is one of the most popular methods for stereo matching. In the era of deep learning, people leveraged deep neural networks to generate the cost volume (Zbontar et al., 2016), perform stereo matching (Zhang et al., 2019), or both. These DL methods clearly outperform traditional, non-DL methods on standard stereo benchmarks, such as Middlebury, KITTI, and ETH3D. Recently, more end-to-end DL methods emerged. These methods replace the classic two-stage pipeline with a single neural network architecture that takes in a rectified image pair and directly outputs the disparity map. These end-to-end DL methods have outperformed two-stage DL methods and demonstrated the top performances in computer vision benchmarks (Lipson et al., 2021, Zhao et al., 2022, Li et al., 2022a).

There have been several studies that quantitatively analyze the performance of DL stereo methods on satellite stereo (Albanwan and Qin, 2022, Gómez et al., 2022). Albanwan *et al.* (Albanwan and Qin, 2022) present an extensive comparative study

[1] Lidar DSM coverage

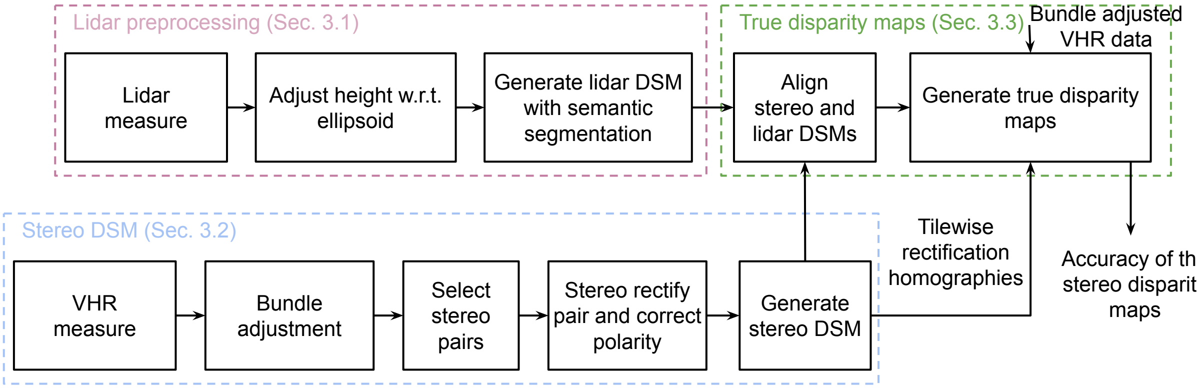

Lidar preprocessing (Sec. 3.1)
Lidar measure
Adjust height w.r.t. ellipsoid
Generate lidar DSM with semantic segmentation
True disparity maps (Sec. 3.3)
Bundle adjusted VHR data
Align stereo and lidar DSMs
Generate true disparity maps
Stereo DSM (Sec. 3.2)
VHR measure
Bundle adjustment
Select stereo pairs
Stereo rectify pair and correct polarity
Generate stereo DSM
Tilewise rectification homographies
Accuracy of the stereo disparity maps

Figure 2. Stellar data generation pipeline.

of various stereo-matching approaches by fine-tuning them using a relatively small satellite stereo dataset (DFC in Table 1). They analyze two-stage approaches including SGM using a non-DL census cost volume (Zabih and Woodfill, 1994) and using a DL-based cost volume (Zbontar et al., 2016), as well as end-to-end DL methods (Kendall et al., 2017, Chang and Chen, 2018, Cheng et al., 2020). They report that the two-stage approaches, whether using DL cost volume or not, are more robust and generalizable. Meanwhile, end-to-end DL methods have the highest geometric accuracy while not generalizing well for unseen data. To increase the robustness of these end-to-end DL methods, a large satellite stereo dataset that covers more diversity in its stereo pairs seems necessary.

2.2 Satellite Stereo Datasets

Table 1 provides a comparison among previous satellite stereo datasets and Stellar. A critical challenge in creating a satellite stereo dataset is the generation of ground truth disparity. Typically, people leverage Lidar measurements to synthesize the ground truths, but the Lidar point clouds are sparse and are in different coordinate systems than satellite imagery. The Multi-View Stereo 3D Mapping dataset (MVS3DM) (Bosch et al., 2016) only contains Lidar point clouds as ground truths and the authors do not calibrate the Lidar data and the Satellite imagery into the same coordinate system, which could cause difficulty in performing supervised training using the dataset. The 2019 Data Fusion Contest dataset (DFC) (Bosch et al., 2019) includes sparse true disparity maps generated from the Lidar point clouds. Its Lidar measurements and satellite imagery are aligned by maximizing mutual information between the two sources of data. SatStereo (Patil et al., 2019a) contains dense ground truths. The authors first create true DSMs from the Lidar measurements, then transform the Lidar DSMs into dense disparity maps in the same coordinate as the satellite imagery via bundle adjustments.

A key difference between the satellite stereo datasets and stereo datasets in the computer vision community, such as Middlebury, KITTI, and ETH3D, is that the disparity maps are naturally bipolar in satellite stereo, meaning the disparity maps contain both positive and negative values. This is caused by the fundamental difference in camera model between the satellite cameras and pinhole cameras. See Sec 3.2.2 for detailed discussions. As in Table 1, all listed previous datasets contain both positive and negative disparity values. Adapting a stereo-matching algorithm to work for the both positive and negative disparity, especially those based on semi-global matching (SGM), is non-trivial. Stellar overcomes this issue. It enables all positive disparity via an innovative tile-based polarity correction (Sec 3.2.3.)

Compared to previous satellite stereo datasets, Stellar enables much higher flexibility. Compared to the recently released WHU-Stereo dataset (Li et al., 2022b), which contains a similar covered area as Stellar (Table 1), Stellar contains more than one stereo pair per region with different baseline distances and acquisition times that users can flexibly choose. This could help an algorithm learn to handle varying baseline distances and different visual appearances within stereo pairs.

3. DATA GENERATION PIPELINE

Figure 2 shows the overview of Stellar data generation pipeline. Key steps include (a) Lidar preprocessing, (b) Stereo DSM, and (c) True disparity maps generation. We explain each step in an individual subsection.

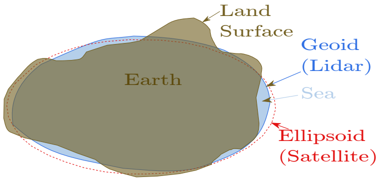


Figure 3. As UAVs that carry Lidars and satellites use different vertical datum, the two data sources must be calibrated to the same coordinate system.

3.1 Lidar Data Processing

Lidars measure a sparse point cloud $\{\mathbf{x}_i, i = 1, \cdots, N\}$ of the terrain. Each point $\mathbf{x_i}$ is a four-vector $\mathbf{x_i} = (x_i, y_i, z_i, c_i)$, where (x_i, y_i) indicates the Lambert conformal conic projection (LCC) coordinate of the point, z_i is the altitude of the point with respect to some zero altitude datum, and c_i is a semantic label of the point, e.g. building, bridge, water, etc., that is sometimes present in the data.

The first step of Lidar data processing is coordinate transformation. The Lidar point clouds follow NAVD88 geoid vertical datum, which uses the approximate mean sea level as the zero altitudes (Figure 3.) Meanwhile, the satellite imagery is collected under a different vertical datum, WGS84 ellipsoid. Therefore, the first step is to convert the altitude of each point z_i into the same vertical datum as the satellite imagery:

$$\tilde{z}_i = \text{Vertical Datum Conversion}(z_i).$$

We achieve the conversion using a predefined transformation grid. We also transform the 2D location of each point from LCC coordinate (x_i, y_i) to Universal Transverse Mercator (UTM) coordinate $(\tilde{x}_i, \tilde{y}_i)$. We denote the point cloud after coordinate transformation as $\{\tilde{\mathbf{x}}_i = (\tilde{x}_i, \tilde{y}_i, \tilde{z}_i, c_i)\}$. For simplicity of notation, we omit the index range $i = 1, 2, \cdots$ hereafter.

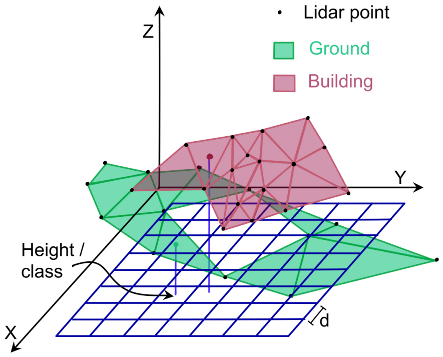


Figure 4. The process of generating dense 2D maps of altitude, i.e., DSM, and semantic classes from a point cloud. First, triangulate the point clouds and prune the redundant triangles to form a mesh. Then, project each pixel in the 2D map to the mesh and determine the value of the pixel, i.e., altitude or semantic class, by interpolating the vertices of the intersecting triangle. See detailed description in Section 3.1.1.

3.1.1 Lidar DSM Generation After the transformation, we rasterize the transformed point cloud $\{\tilde{\mathbf{x}}_i\}$ to a dense 2D altitude map, i.e., the DSM, and a semantic segmentation map. As in Figure 4, we first use Delaunay triangulation to obtain a mesh from the point cloud, $\mathcal{T} = \text{Delaunay}(\{(\tilde{x}_i, \tilde{y}_i, \tilde{z}_i)\})$. The mesh $\mathcal{T}$ contains a list of triangles t_j whose vertices are neighboring points in the point cloud, $\mathcal{T} = \{t_j\}$. Next, we prune the redundant triangles in the mesh $\mathcal{T}$ to form a new mesh $\mathcal{S}$ based on the following rule:

$$\begin{aligned}\mathcal{S} =\{&t_j \in \mathcal{T} \mid \text{All edges of } t_j \text{ are shorter than } \alpha\\ &\textit{AND}\\ &\text{The angle between } t_j\text{'s normal vector}\\ &\text{and all its neighboring triangles' normal vector}\\ &\text{are below } \beta\\ &\textit{AND}\\ &\text{The maximum altitude difference among } t_j\text{'s vertices}\\ &\text{is smaller than } \gamma\}.\end{aligned}$$

We use $\alpha = 10m$, $\beta = 20°$, and $\gamma = 1m$. Then, as in Figure 4, we define a grid with a step size d and project each point in the grid along the z-axis to intersect with the mesh to find out its altitude. If the ray intersects a triangle t_j in the mesh, the height is estimated as the interpolated value of the three vertices of t_j. If a ray doesn't intersect any triangles, that point is marked as invalid. In our experiment, we define $d = 0.3m$ or $0.5m$.

Figure 5 shows a qualitative comparison between the Lidar DSMs in DFC (Bosch et al., 2019) and in Stellar for two regions included in both datasets. The average height difference between the two Lidar DSMs is less than 0.5 m. As highlighted using arrows in Figure 5, Stellar DSMs are more accurate along boundaries such as edges of buildings and include more fine details compared to DFC.

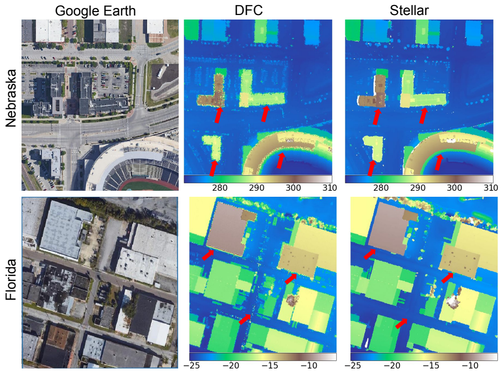
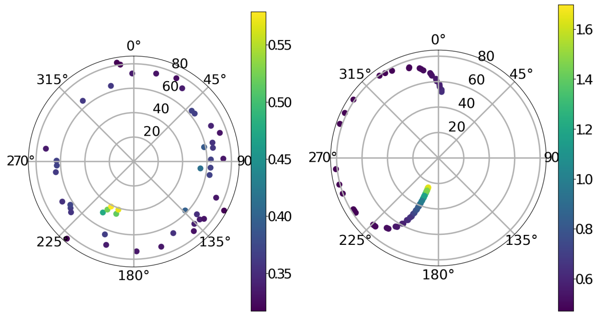


Figure 5. Qualitative comparison of the Lidar DSMs from DFC (Bosch et al., 2019) and Stellar. As highlighted using arrows, Stellar DSMs contain more fine details and are more accurate along building boundaries. (First column image source: ©Google Earth)

3.1.2 Semantic Segmentation As mentioned before, the semantic label may not be present in some point cloud data. In this case, we leverage the connected component analysis on the mesh $\mathcal{S}$ to cluster all triangles. Then we label the largest connected component as ground and the moderate size connected components, *i.e.*, components with at least 500 triangles, as buildings (Figure 4.) The semantic segmentation map is then generated using the same projection method described in Section 3.1.1. A sample semantic segmentation map is in Figure 1(e).



(a) Argentina (WV3) (b) Georgia (WV2)

Figure 6. Satellite positions per image for each region. Outer ring in each plot represents azimuth angle, the inner rings represent off-nadir angles and GSD (meters) for each image.

3.2 Satellite Stereo DSM

Each image **I** from the satellite comes with an approximate camera model $\mathcal{P}$ in rational polynomial coefficient (RPC) format, and metadata such as satellite (Figure 6), sun positions, timestamp, and ground sampling distance (GSD). The image we use is the panchromatic image that has a spatial resolution of 0.3-0.7m.

3.2.1 Data Alignment and Stereo Pair Selection The provided RPC camera model associated with each satellite image has a relative pointing error up to a few meters w.r.t. other images in the same region. Similar to the practice by Patil *et al.* (Patil et al., 2019a), we correct the relative point errors using bundle adjustment. This correction improves the stereo rectification accuracy. Bundle adjustment is the process that, given a set of images $\{\mathbf{I}_1, \cdots, \mathbf{I}_p\}$, determines the corresponding camera projection function of each image $\mathcal{P}_1, \cdots, \mathcal{P}_p$.

First, we use the SIFT/SURF to detect a set of corresponding key points in each image. We denote the ith key point in the jth image as $\mathbf{x}_i^j$. We also perform RANSAC to remove outliers to ensure correspondences among key points. Then, we unproject the ith key point for all images $\{\mathbf{x}_i^j, j = 1, \cdots, p\}$ using the estimated projection functions $\{\mathcal{P}_j, j = 1, \cdots, p\}$ to locate the corresponding 3D world point $\mathbf{X}_i$ via triangulation, and project $\mathbf{X}_i$ back to each view to calculate the reprojection error. We use this reprojection error as an objective function to search for the optimal camera projection functions:

$$\hat{\mathcal{P}}_{1,\cdots,p} = \underset{\mathcal{P}_{1,\cdots,p}}{\arg\min} \sum_{j\neq k} \sum_{i} ||\mathbf{x}_i^j - \mathcal{P}_j(\mathcal{P}_k^{-1}(\mathbf{x}_i^k))||^2. \qquad (1)$$

In multi-date satellite stereo reconstruction, multiple factors such as view difference (Figure 6), sun angle difference, acquisition time difference, etc., influence the quality of DSM reconstruction. Given a set of n satellite images, there are ${}_n\mathrm{C}_2$ possible stereo pairs. We ignore stereo pairs with view angle differences less than 5° to avoid those with narrow baselines. All the rest possible stereo pairs are accessible to users in Stellar.

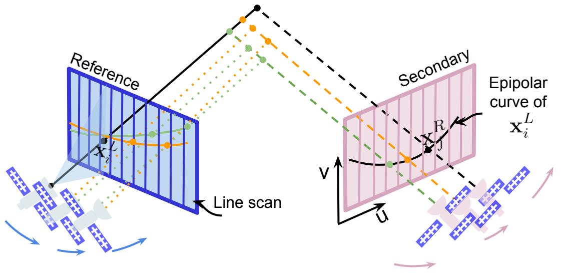


Figure 7. Satellite image sensors are pushbroom cameras, which have non-linear homography and non-conjugate epipolar curves. See detailed discussion in Section 3.2.2.

3.2.2 Epipolar Geometry of Satellite Imagers

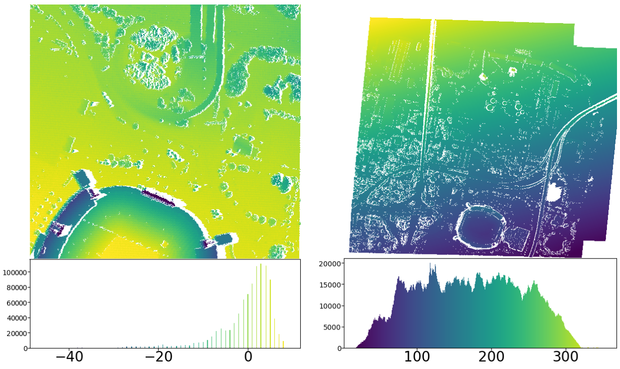
The linear pushbroom (LP) camera used in earth observation satellites has only one line of pixels in its photosensor. As the satellite moves along its orbit, the LP camera scans the earth (Figure 7.) During the scanning, the camera center moves at a constant speed. This has a significant impact on the epipolar geometry of two LP cameras. Meanwhile, cameras in our daily life are usually perspective, which has a unique camera center and a constant projection matrix for each captured image.

For a pair of images captured by LP cameras, there is a different fundamental matrix $\mathbf{F}_{ij}$ for every ith column of the left image and every jth column of the right image. As in Figure 7, for a point $\mathbf{x}_i^L$ on the ith column of the left image, it will correspond to an epipolar curve c_R on the right image (Gupta and Hartley, 1997). The epipolar curve c_R intersects the jth column at $\mathbf{x}_j^R$:

$$\mathbf{x}_j^R = (j, v), \qquad (2)$$

$$\text{where } (u', v', w') = \left(\mathbf{F}_{ij}\mathbf{x}_i^L\right) \times (1, 0, 0), \text{ and } v = v'/w'.$$

Meanwhile, each point $\mathbf{x}_j^R$ corresponds to different epipolar curves on the left image, which means the epipolar curves are non-conjugate. Denote $\mathbf{C}_i^L$ and $\mathbf{C}_j^R$ of the camera center when capturing the ith column of the left image and the jth column of the right image. The baseline vector $\overrightarrow{\mathbf{C}_i^L\mathbf{C}_j^T}$ will vary its direction for different i, j in most situations (Habib et al., 2005). Thus, the disparity between the left and right images could become both positive and negative.



(a) DFC (b) Stellar

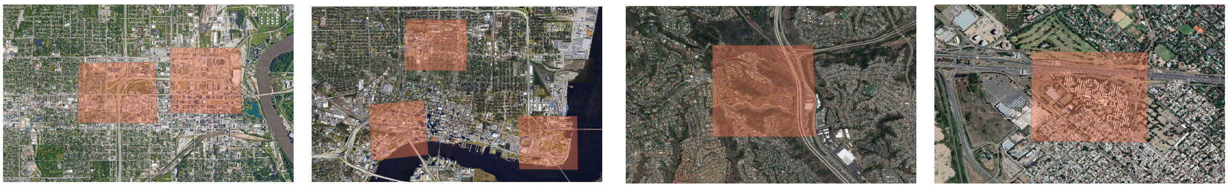
Figure 9. Polarity comparison of true disparity maps. The bottom plots are histograms of disparity. The disparity values of Stellar is always positive.

3.2.3 Stereo Rectification and Polarity Correction

Rectifying a pair of satellite images is non-trivial because of the hyperbolic, non-conjugate epipolar geometry between the images. To deal with this, we divide the satellite images into tiles and assume each tile to follow the affine camera projective geometry. Thus, we can assume linear, conjugate epipolar geometry between each pair of tiles. De Franchis *et al.* also used similar tilewise affine assumptions (De Franchis et al., 2014), but we have several innovations compared to their practice. First, to efficiently identify corresponding tiles between the left and right images, we use the DEM sculpting approach (Patil et al., 2019b) to obtain a rough estimate of correspondences among tiles by using the low-resolution digital elevation model (DEM) of the earth. Second, we modify the rectification homography for the right image to apply additional translation to obtain unipolar disparities.

Given a set of images $\{\mathbf{I}_1, \cdots, \mathbf{I}_p\}$ with significant scene overlap and their corresponding bundle adjusted camera projection functions $\{\hat{\mathcal{P}}_1, \cdots, \hat{\mathcal{P}}_p\}$, we pick a stereo pair $(\mathbf{I}^L, \mathbf{I}^R) \in \{\mathbf{I}_1, \cdots, \mathbf{I}_p\}$ and their camera projection functions $(\mathcal{P}^L, \mathcal{P}^R) \in \{\hat{\mathcal{P}}_1, \cdots, \hat{\mathcal{P}}_p\}$. We divide each image in the selected stereo pair into n 500 pixel × 500 pixel tiles with 100-pixel overlap between neighboring tiles. This gives us a list of tiled stereo pairs $\{(\mathbf{I}_i^L, \mathbf{I}_i^R), i = 1, \cdots, n\}$.

We obtain a set of virtual correspondences in each tiled stereo pair using DEM-Sculpting (Patil et al., 2019b) as follows. The digital elevation model (DEM) is a two-dimensional altitude map of the terrain, similar to DSM but with a much lower spatial resolution. People already have access to the DEM of the entire earth. We uniformly sample points $\{\mathbf{x}_j^L, j = 1, \cdots, K_1\}$ on each left tile $\mathbf{I}_i^L$ and unproject these points onto the $30m$-resolution DEM to obtain the corresponding 3D world point $\mathbf{X}_j = (\mathcal{P}^L)^{-1}(\mathbf{x}_j^L)$. We then project each 3D point $\mathbf{X}_j$ onto the right tile to calculate the corresponding pixel location $\mathbf{x}_j^R = \mathcal{P}^R(\mathbf{X}_j)$. We compute another set of correspondences by creating another set of 3D world points $\tilde{\mathbf{X}}_j = \mathbf{X}_j + (0, 0, 100m)$ and project them onto $\mathbf{I}^L$ and $\mathbf{I}^R$. We ignore any point that falls outside the boundary of the tiles $(\mathbf{I}_i^L, \mathbf{I}_i^R)$. We denote the new set



(a) Training sites in Nebraska. (b) Training sites in Florida. (c) Testing sites in California. (d) Testing sites in Argentina.

Figure 8. Training and testing sites.(Image source ©Google Earth)

of correspondences as $\{(\tilde{\mathbf{x}}_j^L, \tilde{\mathbf{x}}_j^R), j = 1, \cdots, K_2\}$. This new set of points creates an elevation envelope to account for the altitude of buildings, as they do not exist in the low-resolution DEM. In total, the process generates $K_1 + K_2$ virtual correspondences between the left and right tile $(\mathbf{I}_i^L, \mathbf{I}_i^R)$.

Then, we find the rectification homography $\mathbf{H}_i^L$ and $\mathbf{H}_i^R$ for the tile pair. Mathematically, the fundamental matrix $\mathbf{F}_i$ of the two tiles has the following form (Hartley and Zisserman, 2003).

$$\mathbf{F}_i = \begin{bmatrix} 0 & 0 & a \\ 0 & 0 & b \\ c & d & e \end{bmatrix},$$

as we assume affine camera model. Using the previously identified $K = K_1 + K_2$ virtual correspondences, denoted as $(\mathbf{x}_k^L, \mathbf{x}_k^R), k = 1, 2, \cdots, K$ for simplicity, we can estimate the five parameters of the fundamental matrix using the Gold Standard algorithm (Hartley and Zisserman, 2003). The epipoles for left and right images can be given as $\mathbf{e}^L = (-d, c, 0)^T$ and $\mathbf{e}^R = (-b, a, 0)^T$, respectively. We compute the stereo rectification homography as follows:

$$\mathbf{H}_i^L = \begin{bmatrix} s\mathbf{R}^L & \mathbf{t}^L \\ \mathbf{0}^{1\times 2} & 1 \end{bmatrix}, \quad \mathbf{H}_i^R = \mathbf{T}\tilde{\mathbf{H}}_i^R \tag{3}$$

where

$$\tilde{\mathbf{H}}_i^R = \begin{bmatrix} 1/s\mathbf{R}^R & \mathbf{t}^R \\ \mathbf{0}^{1\times 2} & 1 \end{bmatrix}, \mathbf{R}^L = \frac{1}{||\mathbf{e}||} \begin{bmatrix} -d & c \\ -c & -d \end{bmatrix},$$

$$\mathbf{R}^R = \frac{1}{||\mathbf{e}'||} \begin{bmatrix} -b & a \\ -a & -b \end{bmatrix}, \mathbf{T} = \begin{bmatrix} 0 & 0 & -t_x \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

$s = \frac{||\mathbf{e}||}{||\mathbf{e}'||}$, $\mathbf{t}^L = (0, -t)^T$, $\mathbf{t}^R = (0, t)^T$ and $t = \frac{e}{2||\mathbf{e}||||\mathbf{e}'||}$. The matrix $\mathbf{T}$ shifts the right image horizontally by t_x so that the disparity is unipolar. The shift t_x is the maximum disparity of all the K rectified correspondences:

$$t_x = \max\{\mathbf{H}_i^R \mathbf{x}_k^R - \tilde{\mathbf{H}}_i^L \mathbf{x}_k^L, k = 1, 2, \cdots, K\}. \tag{4}$$

With the homography, we can obtain rectified tile pairs $\{(\tilde{\mathbf{I}}_i^L, \tilde{\mathbf{I}}_i^R)\}$. Then, we stitch all the rectified pairs together to form two rectified images $(\tilde{\mathbf{I}}^L, \tilde{\mathbf{I}}^R)$. In case of overlapping pixel value for multiple tiles, we pick the value from the tile with a smaller index i. For simplicity, we denote the entire rectification process as $\tilde{\mathbf{I}}^{L/R} = g^{L/R}(\mathbf{I}^{L/R})$, where $\tilde{\mathbf{I}}^{L/R}$ is the rectified left/right image and $g^{L/R}$ is the left/right rectification function.

Figure 9 shows a comparison of the disparity maps and their histograms for DFC and Stellar datasets. It is clear that Stellar contains all positive disparities.

3.2.4 DSM Generation from Disparity Maps

In this section, we discuss an intermediate DSM generated from satellite images. This DSM is for aligning the Lidar data with the satellite data. Given a rectified stereo pair $(\tilde{\mathbf{I}}^L, \tilde{\mathbf{I}}^R)$, we use the modified tSGM algorithm (Patil et al., 2019b) to output two disparity maps $(\mathbf{D}^L, \mathbf{D}^R)$, where $\mathbf{D}^L$ is the left disparity and $\mathbf{D}^R$ is the right disparity. For a point $\mathbf{x}^L$ in the left image $\mathbf{I}^L$, we can use the following transformation to find the corresponding point on the right image $\mathbf{I}^R$:

$$\mathbf{x}^R = (g^R)^{-1} \left(\mathbf{D}^L(g^L(\mathbf{x}^L)) \right).$$

By unprojecting points $\mathbf{x}^L$ and $\mathbf{x}^R$ using their corresponding camera projection functions and using triangulation, we can locate their corresponding 3D world point $\mathbf{X}$. We repeat this process for all points in the left and right images and can generate a 3D point cloud of the scene. Then we can use the same triangulation and interpolation process in Section 3.1.1 to produce a DSM.

3.3 Ground Truth Disparity Map Generation

The calibration of the Lidar DSM and the alignment DSM described in Section 3.2.4 follows the procedure in Nuth *et al.* (Nuth and Kääb, 2011). It minimizes the overall elevation differences between the two DSMs by performing a global translation and bias of the Lidar DSM.

After alignment, we project all points $\mathbf{X}$ from the Lidar DSM to the left and right rectified image via $\tilde{\mathbf{x}}^{L/R} = g^{L/R}(\mathcal{P}^{L/R}(\mathbf{X}))$ and calculate the true left disparity as the horizontal difference between the projected points: $\tilde{\mathbf{D}}^L(\tilde{\mathbf{x}}^L) = \tilde{\mathbf{x}}_u^L - \tilde{\mathbf{x}}_u^R$. We then treat $\tilde{\mathbf{D}}^L$ as a point cloud and use the triangulation and interpolation approach in Section 3.1.1 to obtain the ground truth of left disparity map $\mathbf{D}^L$. We repeat the same process to obtain $\mathbf{D}^R$. Then, we perform a left-right-right-left (LRRL) consistency check to remove pixels where both disparity values disagree, which could happen at occlusion. Mathematically, the LRRL check is:

$$\mathbf{D}^L(\mathbf{x}) = \begin{cases} \mathbf{D}^L(\mathbf{x}) & if \quad |\mathbf{D}^L(\mathbf{x}) - \mathbf{D}^R(\mathbf{s})| \leq 1 \\ invalid & otherwise \end{cases} \tag{5}$$

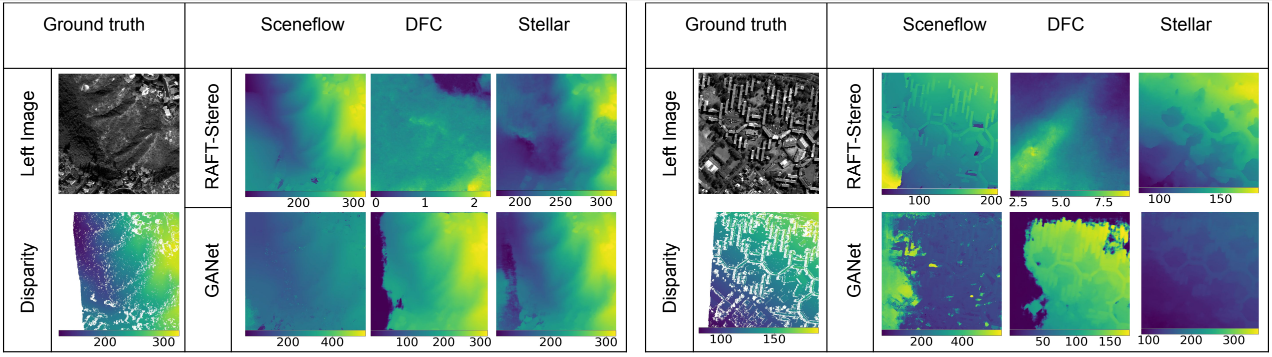
where $\mathbf{s} = (\mathbf{x}_u - \mathbf{D}^L(\mathbf{x}), \mathbf{x}_v)$ and $\mathbf{x}$ is the position of a pixel in the left image.

4. PRELIMINARY EXPERIMENTAL RESULTS

This section describes our preliminary effort to analyze the effectiveness of Stellar. We select two pre-trained DL stereo architectures, fine-tune them on Stellar and DFC (Bosch et al., 2019) datasets, and analyze their performance. The information about the two DL architectures is as follows:

GANet (Zhang et al., 2019) follows the classic two-stage stereo matching pipeline. It first extracts dense features from the input image pair using a convolutional architecture to compute a

Method	California			Argentina		
	Good 3 (%) ↑	Avg (px) ↓	RMSE (px) ↓	Good 3 (%) ↑	Avg (px) ↓	RMSE (px) ↓
GANet (Sceneflow)	66.86	39.90	64.17	37.34	72.60	111.54
GANet (DFC)	70.36	71.32	103.61	28.27	141.18	174.24
GANet (Stellar)	71.96	25.41	36.05	48.74	37.9	46.32
RAFT-Stereo (Sceneflow)	84.48	16.76	26.32	48.61	47.27	62.83
RAFT-Stereo (DFC)	0.20	347.02	349.42	3.17	256.24	261.17
RAFT-Stereo (Stellar)	59.00	43.68	50.53	49.7	30.86	34.89



(a) California

(b) Argentina

Figure 10. Preliminary quantitative (table) and qualitative (figure) analysis of two DL methods for satellite stereo. **Table.** Best and second best values are highlighted for each metric. Methods fine-tuned on Stellar generally perform the best on both testing regions, while RAFT-Stereo pretrained on Sceneflow performs the best on the California region. **Figures.** Visually, RAFT-Stereo pre-trained on Sceneflow preserves the best quality of fine details.

cost volume. Then, the network uses semi-global aggregation (SGA) layers and local guided aggregation (LGA) layers to perform an approximated semi-global matching procedure that is differentiable and computationally efficient. The whole architecture is differentiable for end-to-end training.

RAFT-Stereo (Lipson et al., 2021) is an end-to-end approach that directly estimates the disparity map from an image pair without generating the cost volume. The model first extracts feature maps from the image pair to build a multi-resolution correlation pyramid. Then it uses a recurrent architecture, the gated recurrent units (GRUs), to create and iteratively refine the output disparity map.

In our experiment, we use models pre-trained by the original authors on the Sceneflow dataset (Mayer et al., 2016) for both architectures. We fine-tune these pre-trained models on DFC and on a portion of Stellar separately, and test all models on a different portion of Stellar. Figure 8 shows the division of Stellar dataset for this experiment. We use image pairs of two cities as the training set and those of another two as the testing set. The testing images may contain different terrains than the training set. For example, the testing images of California in Figure 8c contain hills that are not present in the training images. Therefore, this experiment also analyzes the generalizability of the models.

The loss function for fine-tuning is:

$$\sum_i \mathcal{L}(\mathbf{D}_{gt}, f(\tilde{\mathbf{I}}_i^L, \tilde{\mathbf{I}}_i^R)), \tag{6}$$

where $\mathbf{D}_{gt}$ is the ground truth disparity map, $f = \{\text{GA-Net}, \text{RAFT-Stereo}\}$, $\{\tilde{\mathbf{I}}_i^L, \tilde{\mathbf{I}}_i^R\}$ are the ith rectified image pair in the training set. GANet use Huber loss and RAFT-Stereo use $\mathcal{L}_1$ loss as the loss function $\mathcal{L}$. The learning rate is 5×10^{-5} for RAFT-Stereo and 10^{-4} for GANet.

We adopt the following metrics for the quantitative evaluation of estimated disparity maps.

- **Good 3**: given as percentage of the valid pixels in $\mathbf{D}_{gt}$ such that $|\mathbf{D}_L(\mathbf{x}) - \mathbf{D}_{gt}(\mathbf{x})| < \delta$ for $\delta = 3$ where $\mathbf{x} = (u, v)^T$ is pixel position in a disparity map and $\mathbf{D}_L$ is an estimated left disparity map. For this metric, we consider all valid pixels in the ground truth disparity map, and it measures the density of estimated disparity maps.

- **Average Error**: Average error between estimated and ground truth disparity map considering valid pixels in both maps.

- **RMSE**: Root Mean Square Error (RMSE) between estimated and ground truth disparity map considering valid pixels in both maps.

Figure 10 shows the quantitative and qualitative evaluation of the two DL models. For GANet, the model fine-tuned with Stellar achieves the best performance in both testing areas. For RAFT-Stereo, the Stellar model performs the best on one testing region, while the pre-trained model achieves the highest accuracy on another. It is worth noticing that both models trained on DFC perform the worst for both regions, which is probably because the ground truth disparity of DFC is sparse and bipolar, and there is limited training data.

It is not crystal clear to us why the pre-trained RAFT-Stereo sometimes performs better than the same architecture fine-tuned on Stellar. A possible explanation is the training data for Stellar is still not sufficient for such sophisticated DL architectures. We envision more comprehensive analyses of DL satellite stereo methods using datasets such as Stellar in future studies.

5. ACKNOWLEDGEMENT

The authors thank Charles Brookshire for coming up with the name Stellar for our dataset.

REFERENCES

Albanwan, H., Qin, R., 2022. A Comparative Study on Deep-Learning Methods for Dense Image Matching of Multi-angle and Multi-date Remote Sensing Stereo Images. *arXiv preprint arXiv:2210.14031.*

Bosch, M., Foster, K., Christie, G., Wang, S., Hager, G. D., Brown, M., 2019. Semantic stereo for incidental satellite images. *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, IEEE, 1524–1532.

Bosch, M., Kurtz, Z., Hagstrom, S., Brown, M., 2016. A multiple view stereo benchmark for satellite imagery. *2016 IEEE Applied Imagery Pattern Recognition Workshop (AIPR)*, IEEE, 1–9.

Brown, M., Goldberg, H., Foster, K., Leichtman, A., Wang, S., Hagstrom, S., Bosch, M., Almes, S., 2018. Large-scale public lidar and satellite image data set for urban semantic labeling. *Laser Radar Technology and Applications XXIII*, 10636, SPIE, 154–167.

Chang, J.-R., Chen, Y.-S., 2018. Pyramid stereo matching network. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5410–5418.

Cheng, X., Zhong, Y., Harandi, M., Dai, Y., Chang, X., Li, H., Drummond, T., Ge, Z., 2020. Hierarchical neural architecture search for deep stereo matching. *Advances in Neural Information Processing Systems*, 33, 22158–22169.

De Franchis, C., Meinhardt-Llopis, E., Michel, J., Morel, J.-M., Facciolo, G., 2014. An automatic and modular stereo pipeline for pushbroom images. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences.*

Gómez, A., Randall, G., Facciolo, G., von Gioi, R. G., 2022. An experimental comparison of multi-view stereo approaches on satellite images. *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 844–853.

Gupta, R., Hartley, R. I., 1997. Linear pushbroom cameras. *IEEE Transactions on pattern analysis and machine intelligence*, 19(9), 963–975.

Habib, A. F., Morgan, M., Jeong, S., Kim, K.-O., 2005. Analysis of epipolar geometry in linear array scanner scenes. *The Photogrammetric Record*, 20(109), 27–47.

Hartley, R., Zisserman, A., 2003. *Multiple view geometry in computer vision*. Cambridge university press.

Hirschmuller, H., 2005. Accurate and efficient stereo processing by semi-global matching and mutual information. *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, 2, IEEE, 807–814.

Kendall, A., Martirosyan, H., Dasgupta, S., Henry, P., Kennedy, R., Bachrach, A., Bry, A., 2017. End-to-end learning of geometry and context for deep stereo regression. *Proceedings of the IEEE international conference on computer vision*, 66–75.

Li, J., Wang, P., Xiong, P., Cai, T., Yan, Z., Yang, L., Liu, J., Fan, H., Liu, S., 2022a. Practical stereo matching via cascaded recurrent network with adaptive correlation. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 16263–16272.

Li, S., He, S., Jiang, S., Jiang, W., Zhang, L., 2022b. WHU-Stereo: A Challenging Benchmark for Stereo Matching of High-Resolution Satellite Images. *arXiv:2206.02342*. https://arxiv.org/abs/2206.02342.

Lipson, L., Teed, Z., Deng, J., 2021. Raft-stereo: Multilevel recurrent field transforms for stereo matching. *2021 International Conference on 3D Vision (3DV)*, IEEE, 218–227.

Mayer, N., Ilg, E., Häusser, P., Fischer, P., Cremers, D., Dosovitskiy, A., Brox, T., 2016. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*. arXiv:1512.02134.

Menze, M., Geiger, A., 2015. Object scene flow for autonomous vehicles. *Conference on Computer Vision and Pattern Recognition (CVPR).*

Nuth, C., Kääb, A., 2011. Co-registration and bias corrections of satellite elevation data sets for quantifying glacier thickness change. *The Cryosphere*, 5(1), 271–290.

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

Patil, S., Prakash, T., Comandur, B., Kak, A., 2019b. A comparative evaluation of SGM variants (including a new variant, tMGM) for dense stereo matching. *arXiv preprint arXiv:1911.09800.*

San Diego LiDAR Report, n.d. `https://noaa-nos-coastal-lidar-pds.s3.amazonaws.com/laz/geoid18/8614/supplemental/USGS_Eastern_San_Diego_County_LiDAR_ProjectReport.pdf`.

Scharstein, D., Hirschmüller, H., Kitajima, Y., Krathwohl, G., Nešić, N., Wang, X., Westling, P., 2014. High-resolution stereo datasets with subpixel-accurate ground truth. *German conference on pattern recognition*, Springer, 31–42.

Van Etten, A., Lindenbaum, D., Bacastow, T. M., 2018. Spacenet: A remote sensing dataset and challenge series. *arXiv preprint arXiv:1807.01232.*

Zabih, R., Woodfill, J., 1994. Non-parametric local transforms for computing visual correspondence. *Computer Vision-ECCV'94: Third European Conference on Computer Vision, Stockholm, Sweden, May 2-6, 1994. Proceedings*, 2, Springer Science & Business Media, 151.

Zbontar, J., LeCun, Y. et al., 2016. Stereo Matching by Training a Convolutional Neural Network to Compare Image Patches. *Journal of Machine Learning Research*, 17(1-32), 2.

Zhang, F., Prisacariu, V., Yang, R., Torr, P. H., 2019. Ga-net: Guided aggregation net for end-to-end stereo matching. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).*

Zhao, H., Zhou, H., Zhang, Y., Zhao, Y., Yang, Y., Ouyang, T., 2022. Eai-stereo: Error aware iterative network for stereo matching. *Proceedings of the Asian Conference on Computer Vision*, 315–332.

This contribution has been peer-reviewed.