A Multiple View Stereo Benchmark for Satellite Imagery

Marc Bosch, Zachary Kurtz, Shea Hagstrom, and Myron Brown
The Johns Hopkins University Applied Physics Laboratory
Laurel, Maryland

Abstract **–** The availability of public multiple view stereo benchmark datasets has been instrumental in enabling research to advance the state of the art in the field and to apply and customize methods to real-world problems. Until now, no public multiple view stereo benchmark dataset has been available for satellite imaging applications. In this work, we describe a public benchmark dataset for multiple view stereo applied to three-dimensional outdoor scene mapping using commercial satellite imagery. This dataset includes fifty Digital Globe WorldView-3 panchromatic and multispectral images of a 100 square kilometer area near San Fernando, Argentina. We also provide high-resolution airborne lidar ground truth data for a 20 square kilometer subset of this area and performance analysis software to assess accuracy and completeness metrics. We report initial results from available solutions using this benchmark data and encourage continued research by making this benchmark dataset publicly available to the research community.

I. INTRODUCTION

Earth observation satellite constellations provide global imaging capabilities to support agricultural and environmental monitoring, urban planning, disaster relief, mapping, and defense applications. Multiple view satellite imagery enables estimation of the three-dimensional (3D) structure of a scene, providing a global source of data to support 3D mapping and volumetric change detection. Multispectral imagery further enables characterization of land use and material identification. Until recently, little satellite imagery has been available to the public academic community to support computer vision research. With the rapid pace of satellite imagery constellation development and commercialization, governments and companies have begun to invest in making these data more accessible (e.g., the Planet Labs Open California initiative [1], Landsat on Amazon Web Services [2], and SpaceNet [3]). In this work, we provide a public benchmark dataset for multiple view stereo (MVS) research, development, and metric assessment using commercial satellite imagery. While our focus is MVS, we expect these data to also be useful for a broad range of topics in computer vision and geospatial intelligence.

MVS datasets have been instrumental in encouraging new research and enabling significant improvements in the state of the art. Standardized benchmarks for 3D reconstruction performance are important for evaluating algorithms, and several options are currently available to the community. Each benchmark offers unique features and challenges, with some, like the Middlebury Mview [4], including a full suite of tools, data, ground truth, and recommended metrics. Others include both data and ground truth obtained by an active sensor [19], and some include data alone that has carefully been chosen and collected [6, 21]. These datasets offer an adequate variety of objects and contexts including imagery and ground truth for small indoor objects [4, 5], large outdoor objects [6], and autonomous driving [7]. Benchmark web sites that maintain leaderboards further encourage rigorous testing of new research ideas to quantify their impact [8, 9]. Having several datasets encourages algorithms that generalize well against different challenges.

The proposed benchmark was developed for the IARPA Multi-View Stereo 3D Mapping Challenge [10]. It is being released to the public to encourage greater interest in geospatial data within a broad community of researchers including the conventional fields of photogrammetry and remote sensing and also computer vision and machine learning.

II. MULTIPLE VIEW STEREO

The goal of 3D reconstruction is to estimate the most likely 3D geometry of the scene imaged from multiple viewpoints [37]. Within the computer vision community there has been extensive research in this area, including [33-38]. Recently, with the proliferation of commercial unmanned aerial vehicles and improvement of digital airborne and spaceborne cameras, some research groups have started developing large-scale benchmarks [22, 41] and algorithms [15, 32, 39, 40].

There are two broad categories of approach for multiple view 3D reconstruction from satellite imagery. The first is a straightforward extension of binocular stereo, with many pairs of images processed independently and then the results fused. The second general approach solves the N-view triangulation problem for all images simultaneously.

In pairwise multiple view stereo, pairs of images are processed separately by finding dense correspondences and triangulating their positions using the sensor camera model. A high-quality general-purpose sensor model that minimizes errors when mapping between 3D and the image plane is a

978-1-5090-3284-6/16/$31.00 ©2016 IEEE

critical component of this process, and obtaining a good sensor model is one of the main obstacles to accurate geometry reconstruction. The work of Tao [11] shows that a rational polynomial coefficients (RPC) camera model is one solution to this problem, and demonstrates the inverse mapping and 3D reconstruction process on a stereo image pair. Other variants have built on Tao's method, and multiple examples have been reported [12-15].

Global methods for N-view 3D reconstruction employ a volumetric approach [16, 20] that represents a scene with voxels and use ray casting to derive a probabilistic model describing the presence of a surface within each voxel [17]. Pairwise and volumetric reconstruction methods are compared in [18], highlighting the challenges of developing a robust global solution. For the benchmark dataset described in this paper, an example pairwise algorithm has been implemented and the source code provided to demonstrate basic concepts.

III. BENCHMARK DATASET

The proposed multiple view stereo benchmark for satellite imagery provides images, ground truth lidar data, and several specific challenge areas for metric analysis.

A. Source Imagery

Source imagery for this benchmark dataset is provided courtesy of Digital Globe. Fifty WorldView-3 panchromatic, visible, and near infrared (VNIR) images collected between November 2014 and January 2016 and one short wave infrared (SWIR) image collected in November 2015 are provided in National Imagery Transmission Format (NITF). The area imaged is near San Fernando, Argentina and is shown in Figure 1. Panchromatic image ground sample distance (GSD) is approximately 30cm, VNIR GSD is approximately 1.3m, and SWIR GSD is approximately 4m.

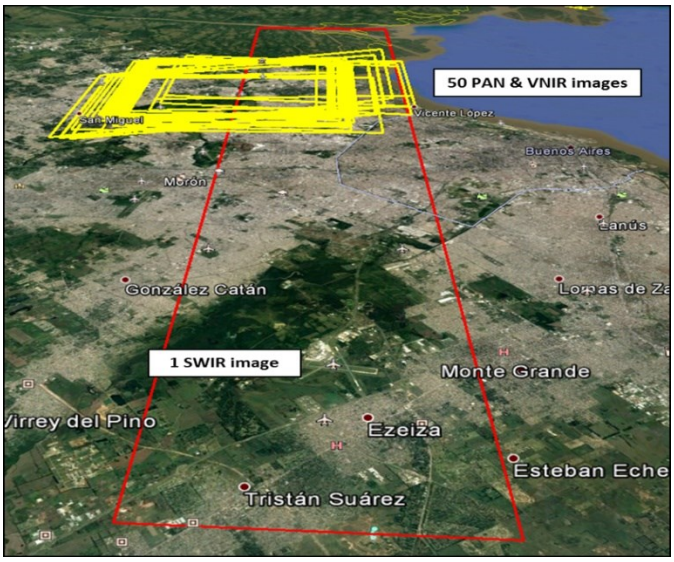

Figure 1. Polygons for satellite imagery coverage provided in the benchmark dataset shown in Google Earth.

B. Ground Truth Lidar

Airborne lidar was collected June 2016 for approximately 20 square kilometers overlapping image coverage as shown in Figure 2. Nominal point spacing is 20 centimeters to support production of 30cm gridded Digital Elevation Model (DEM) products for metric comparison with MVS point clouds generated using the 30cm panchromatic imagery.

Example subsets of the collected lidar data are shown in Figure 3, indicating some areas of interest for metric analysis. These areas include a broad range of urban building heights and spacing for assessing MVS algorithms. Point clouds are RGB colored by height with red high and blue low. Another example is shown in Figure 4, demonstrating the ability of airborne lidar to capture fine detail on vertical features and the significant challenge for doing the same with overhead imagery from space.

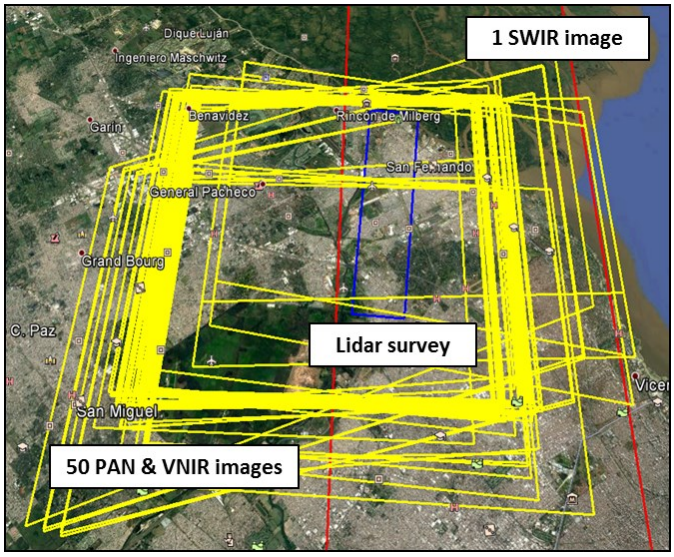

Figure 2. Lidar survey coverage (blue polygon) shown in Google Earth with satellite image coverage polygons.

C. Challenge Datasets

Multiple smaller areas within the overlapping image and lidar extents have been identified and provided in KML files to support metric analysis. Cropped TIFF images and ASCII text metadata files are provided for each of these areas and are generally easier to work with than the much larger full NITF image files.

D. Software

Along with satellite imagery and ground truth lidar data, the benchmark package includes source code to make working with satellite imagery more accessible and to support metric analysis of results. The following sections discuss working with satellite image data and outline the software provided.

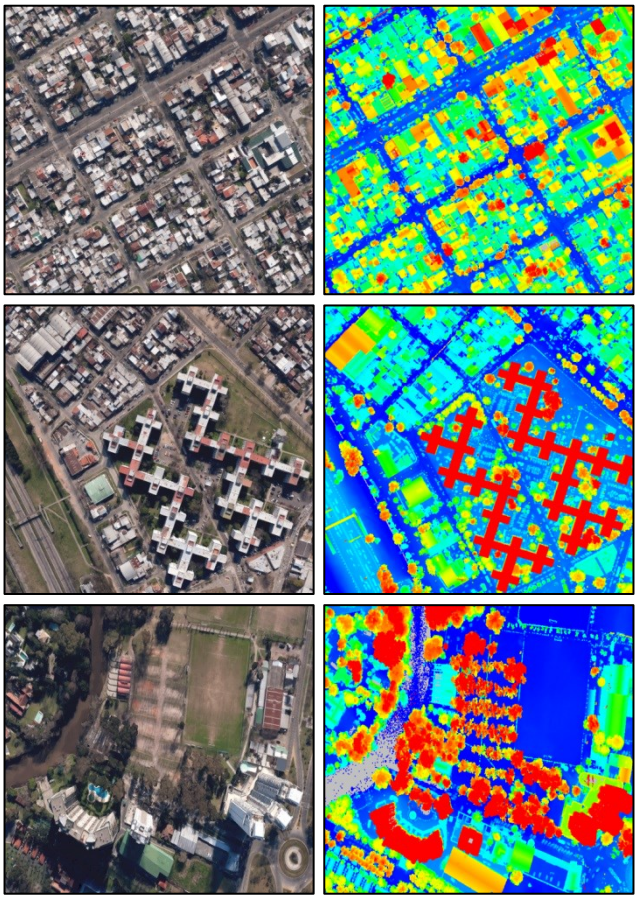

Figure 3. Lidar point clouds shown colored by height with red high and blue low (right) and context imagery shown from Google Earth with 3D terrain option (left).

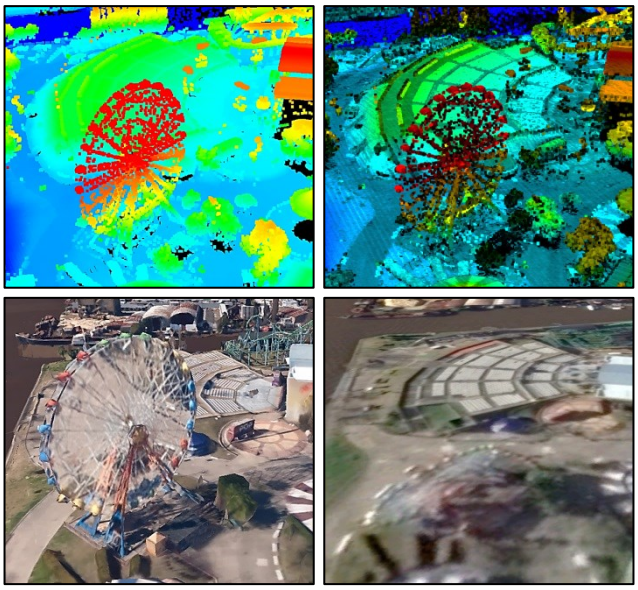

Figure 4. Comparison of lidar and imagery for Ferris wheel highlights limitation of satellite imagery for capturing vertical features; height color lidar point cloud (top left), lidar with intensity (top right), Google Earth 3D terrain based on multiple view airborne imagery (bottom left), and Google Earth imagery (bottom right).

IV. SATELLITE IMAGE CROPPING

To simplify working with very large satellite images, we provide source code for cropping a NITF image and exporting a cropped TIFF image and text metadata file using a KML polygon with the latitude and longitude coordinates of the desired bounding box, as shown in Figure 5.

NITF files contain the parameterized camera model (RPC coefficients) in their header datastream. As part of the metadata accompanying the TIFF files, the RPC coefficients are listed.

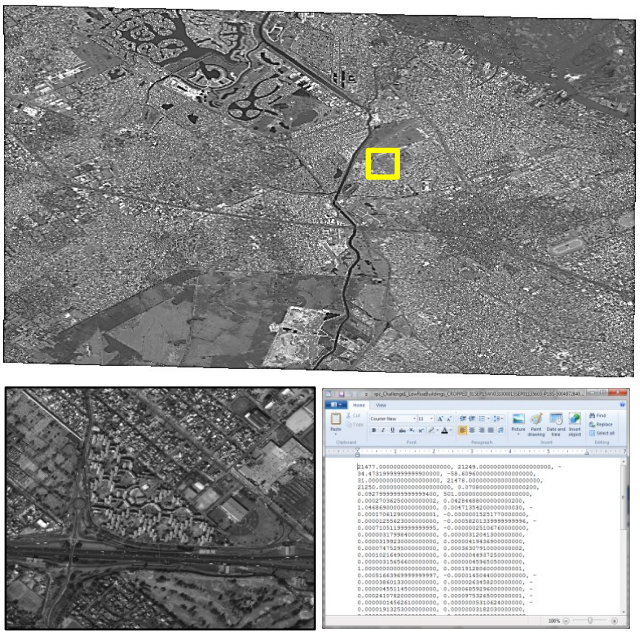

Figure 5. Image cropping software provided in the benchmark package reads a full NITF image (top) and exports a cropped TIFF image (bottom left) and text metadata file for a region identified in a KML polygon.

V. RPC SENSOR MODEL

One challenge in 3D reconstruction given a collection of satellite images is the design of an accurate sensor model. While a direct physics-based sensor model may be ideal for any individual sensor, a generalized model such as the rational function model (RFM) [23] is desirable to support the variety of available sensors. RFM describes intrinsic and extrinsic sensor properties and models them using rational function coefficients (RFC)

$$l_n = \frac{p1(X,Y,Z)}{p2(X,Y,Z)}$$
$$s_n = \frac{p3(X,Y,Z)}{p4(X,Y,Z)} \quad (1)$$

where l_n and s_n represent row and column coordinates in the normalized image coordinate system, X, Y, Z represents the 3D position of the imaged object (*i.e.* Longitude, Latitude, Elevation). $p1(.), p2(.), p3(.)$, and $p4(.)$ are polynomials with RFC coefficients. These are also referred to as rational polynomial coefficients (RPC). RPC is a commonly used replacement sensor model, and it is a specific case of the RFM that is in forward form (mapping of 3D points to pixels) and has third-order polynomials [24]. A tagged record extension for the RPC sensor model is included NITF image file format [25] standard. The following discussion outlines forward and inverse mapping using RPC. Example source code is provided in the benchmark software.

A. RPC Forward Mapping

The forward model of a RPC camera maps 3D points into image plane coordinates. The model starts with the scaling of the world space coordinates to *(-1, 1)* range:

$$\begin{aligned} x &= \frac{X - x_o}{x_s} \\ y &= \frac{Y - y_o}{y_s} \\ z &= \frac{Z - z_0}{z_s} \end{aligned} \tag{2}$$

where X, Y, Z represent longitude, latitude, elevation magnitudes, and $x_o, y_o, z_o, x_s, y_s, z_s$ are offset and scale parameters respectively. They are available as metadata in the NITF image format.

The ratio of two third order polynomials is then calculated to obtain the normalized image coordinates l, s -line (row) and sample (column) according to:

$$\begin{aligned} l_n &= \frac{\sum_{i=1\ldots20} C_{l_i} p1(x,y,z)_i}{\sum_{i=1\ldots20} D_{l_i} p2(x,y,z)_i} \\ s_n &= \frac{\sum_{i=1\ldots20} C_{s_i} p3(x,y,z)_i}{\sum_{i=1\ldots20} D_{s_i} p4(x,y,z)_i} \end{aligned} \tag{3}$$

with

$$\begin{aligned} \sum_{i=1\ldots20} C_{l_i} p1(x,y,z)_i = \\ = C_1 + C_2 x + C_3 y + C_4 z + C_5 xy \\ + C_6 xz + C_7 yz + C_8 x^2 + C_9 y^2 \\ + C_{10} z^2 + C_{11} xyz + C_{12} x^3 + C_{13} xy^2 \\ + C_{14} xz^2 + C_{15} x^2 y + C_{16} y^3 \\ + C_{17} yz^2 + C_{18} x^2 z + C_{19} y^2 z \\ + C_{20} z^3 \end{aligned} \tag{4}$$

Finally, line and sample values are obtained from de-normalizing equation 3:

$$\begin{aligned} l &= l_n \cdot L_s + L_o \\ s &= s_n \cdot S_s + S_o \end{aligned} \tag{5}$$

B. Inverse Mapping Deduction

When only forward mapping coefficients are available, inverse mapping has to be performed from other information sources. Tao et al. showed in [11] that given a stereo pair and their point correspondences, triangulation principles can be used to estimate the 3D point measurement. They show that linearization of the RPC model makes the problem more tractable. Given a point correspondence $(s, l)^{(1)}$ in image 1 and $(s, l)^{(2)}$ in image 2, we can re-write the first order Taylor approximation of the model as:

$$\begin{bmatrix} err_{l_n}^{(1)} \\ err_{s_n}^{(1)} \\ err_{l_n}^{(2)} \\ err_{s_n}^{(2)} \end{bmatrix} = \begin{bmatrix} \frac{\partial_Z l_n^{(1)}}{z_s} & \frac{\partial_Y l_n^{(1)}}{y_s} & \frac{\partial_X l_n^{(1)}}{x_s} \\ \frac{\partial_Z s_n^{(1)}}{z_s} & \frac{\partial_Y s_n^{(1)}}{y_s} & \frac{\partial_X s_n^{(1)}}{x_s} \\ \frac{\partial_Z l_n^{(2)}}{z_s} & \frac{\partial_Y l_n^{(2)}}{y_s} & \frac{\partial_X l_n^{(2)}}{x_s} \\ \frac{\partial_Z s_n^{(2)}}{z_s} & \frac{\partial_Y s_n^{(2)}}{y_s} & \frac{\partial_X s_n^{(2)}}{x_s} \end{bmatrix} \times \begin{bmatrix} \Delta\hat{Z} \\ \Delta\hat{Y} \\ \Delta\hat{X} \end{bmatrix} + \begin{bmatrix} l_n^{(1)} - \hat{l}_n^{(1)} \\ s_n^{(1)} - \hat{s}_n^{(1)} \\ l_n^{(2)} - \hat{l}_n^{(2)} \\ s_n^{(2)} - \hat{s}_n^{(2)} \end{bmatrix} \tag{6}$$

or in matrix form,

$$\boldsymbol{v} = \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b} \tag{7}$$

The least square solution is given by:

$$\boldsymbol{x} = [\Delta Z \ \ \Delta Y \ \Delta X] = (\boldsymbol{A^T W A})^{-1}(\boldsymbol{A^T W b}) \tag{8}$$

with $\boldsymbol{W}$ being the weight matrix for the image points. If some correspondences in the dense matching process are more reliable than others, $\boldsymbol{W}$ can be used to adjust appropriate reliability factors. However, in many cases it is set to the identity matrix. Vector $\boldsymbol{v}$ is iteratively minimized by finding x_k according to equation 8. Initialization can be done by only using the first-order terms of the RPC model, meaning that the first iteration only assumes polynomials with four coefficients. Further refinement using non-linear optimization from the linearized solution leads to slightly more accurate results with an extra computation penalty.

VI. EXAMPLE MVS IMPLEMENTATION

This section describes an algorithm we designed as a baseline for the benchmark evaluation, as well as for testing the dataset resources such as metrics, point cloud registration, and ground truth validation. Figure 6 shows the main steps of the algorithm.

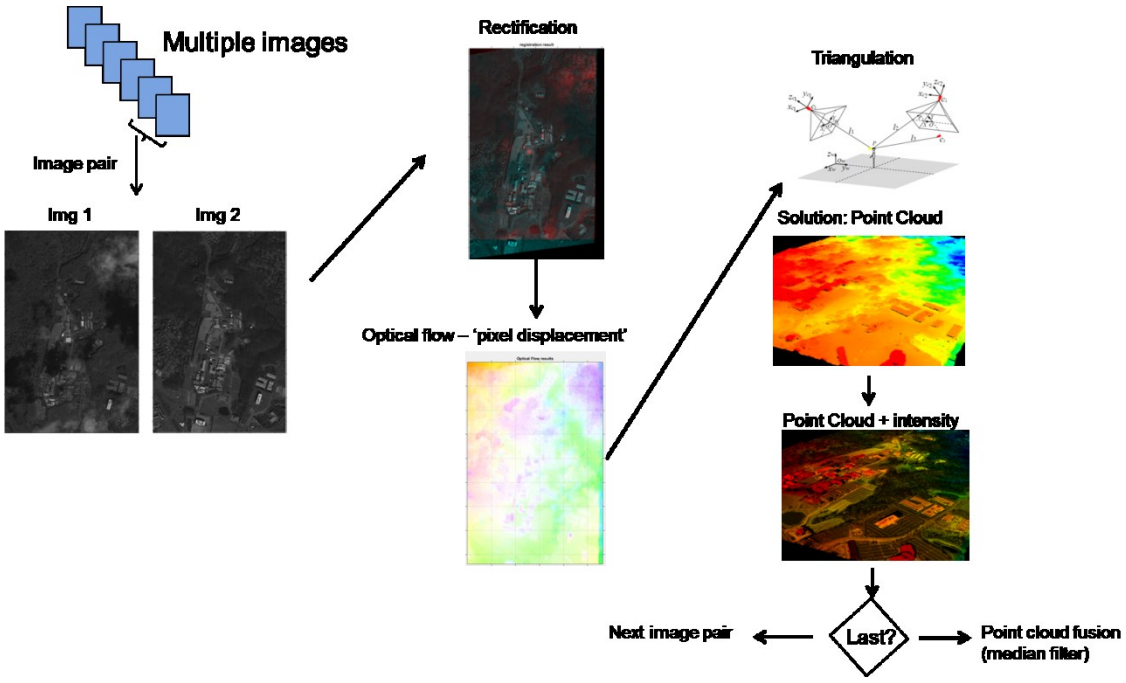

Figure 6. MVS example implementation block diagram.

We designed a baseline algorithm that follows a classic approach of (sparse and/or dense) feature correspondence and point triangulation to generate point clouds. Given a set of multiview images the algorithm processes each pair of images separately. Each image is (planar) rectified with respect to a common grid using matrices H_1 and H_2. For simplicity, we set $H_1 = I_{3x3}$. H_2 is estimated from a collection of sparse feature correspondences ${x^{(1)}}_{i=1..np}$ and ${x^{(2)}}_{i=1..np}$ according to the model $x_i^{(1)} = H_2 \cdot \mathrm{x_i^{(2)}}$[31], where x_i^j is a 2-d homogeneous point ($[s; l; 1]$) in image j. H_2 is modeled as an 8-parameter transformation, *i.e.* $H_2 = [a\ b\ c; d\ e\ f; g\ h\ 1]$ relates points from image 1 and 2. RANSAC robust estimation is used to achieve best rectification [26] given a set of inliers (${x^{(1)}}_{i=1..nin}$ and ${x^{(2)}}_{i=1..nin}$), which are also used for the block adjust step (see figure 6). Once H_2 is found both sets of correspondences are mapped to the reference grid using H_1 according to: $H_1 : x_{ref}^{(1)} = H_1 \cdot x^{(1)} = H_1 \cdot H_2 \cdot x^{(2)}$.

In order to correct misalignments due to parallax and/or noisy estimates of H_2, dense optical flow estimation is performed. In particular, we use SiftFlow algorithm [27] that has proven its efficiency in other applications when following a rectification [28]. We find the planar rectification step critical since it helps reduce the search space of the optical flow and reduce its likelihood of falling into local minima. The combination of both planar rectification and dense SiftFlow gives us the most accurate dense 2-D coordinate match (occlusions aside).

Parallel to the optical flow computation and before the last step (triangulation), we perform block adjustment [11]. It was shown in [29] that simple addition of an offset coefficient into the RPC model can reduce the error of mapping 3-D points back to the image plane. Since the camera sensor is very far from the earth's surface, rays can be assumed to be parallel for each image pixel. Therefore, simple image translations are added into the model described in equation 1. $\hat{s} = \frac{p1(\hat{X},\hat{Y},\hat{Z})}{p2(\hat{X},\hat{Y},\hat{Z})} + \Delta_s$ and $\hat{l} = \frac{p3(\hat{X},\hat{Y},\hat{Z})}{p4(\hat{X},\hat{Y},\hat{Z})} + \Delta_l$ describe the block-adjusted RPC model. Where $\hat{X}, \hat{Y}, \hat{Z}$ are the triangulated points of the subset of inlier points ${x^{(1)}}_{i=1..nin}$ and ${x^{(2)}}_{i=1..nin}$ from the planar rectification step. Δ_s and Δ_l are the translation offsets and are estimated by minimizing the following error measure [29]:

$$error(\Delta_s, \Delta_l) = \sum_{i=1...nin} \left((s,l)_i - \left(\hat{s},\hat{l}\right)_i\right)^2 \quad (9)$$

Once the offsets are estimated, final dense triangulation is done following the model described in section V.B.

When many images are available, reconstruction information from each pair is fused into a final 3D point cloud. In our case, a median filter of the aligned point clouds is performed:

$$PC_{fused}(row, col) = median\begin{pmatrix} PC_{aligned}^{(1)}(row, col), PC_{aligned}^{(2)}(row, col), \dots, \\ PC_{aligned}^{(Npairs)}(row, col) \end{pmatrix}. \quad (10)$$

The example MVS algorithm just described is implemented in MATLAB and uses the cropped TIFF images and text metadata files produced by our NITF cropping tool (section IV).

VII. EFFICIENT SPECTRAL CALIBRATION

WorldView image products are delivered with relative radiometric correction applied to the image data, but many applications require that these digital values be converted to top-of-atmosphere radiance before further processing. Commercial tools are available that use calibration information provided by Digital Globe [30] to perform this absolute calibration, but they are typically very time-consuming. Due to the large number of images in the benchmark dataset, we are providing a software tool and accompanying source code that performs this calibration process significantly faster. Both panchromatic and multispectral WorldView NITF images are supported. The output of this tool is a top-of-atmosphere radiance image in ENVI format. This may be used for RGB encoding of MVS point clouds, though this has not yet been done with the example MVS implementation provided. Figure 7 shows an example of spectral calibration with the benchmark software.

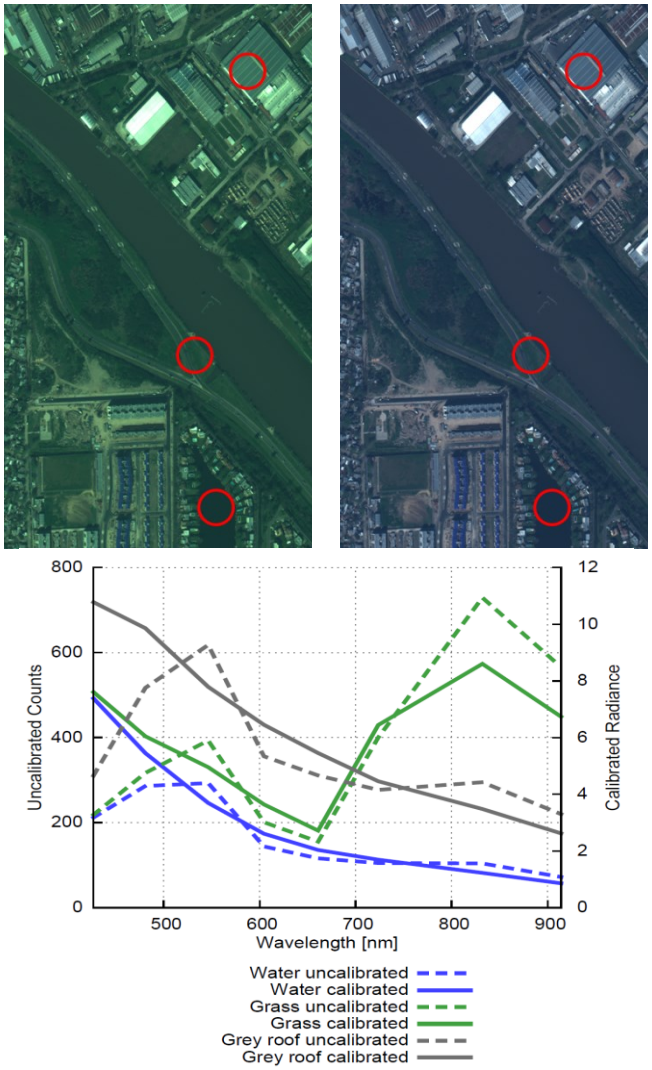

Figure 7. Uncalibrated RGB (left top) shown with calibrated RGB (top right) and spectral samples (bottom).

VIII. EVALUATION METRICS

In this section, we describe the metrics that accompany our dataset. Two metrics are used to evaluate 3D reconstruction fidelity: accuracy and completeness [4]. The metric algorithms evaluate input solutions against truth data to provide an objective measure for comparison.

A. Point Cloud to Grid Conversion

Ground truth data is formatted as a geo-tagged grid of *Z* values covering the challenge area. Each cell of the grid is uniformly spaced across the rectangular extent of the grid.

The input data consists of a geo-tagged point cloud. The input point cloud is converted into an input *Z* grid prior to comparison against the ground truth. The input grid is created by first making a grid of the same size as the truth data. Each point in the input cloud is binned into a grid cell using the point's *X, Y* coordinate. The value for each grid cell is the max *Z* value from all points binned into that cell. If no points land in a particular cell that cell is marked as empty.

B. Accuracy Error

We calculated accuracy error using only the overlapping valid truth and input grid cells. This approach prevents the metric from increasing when the number of empty input cells increases. We account for incomplete input data using the completeness metric (described in section VIII.C).

Algorithm 1 details the accuracy metric. *In* is the input grid being evaluated and *Truth* is the truth *Z* grid. When calculating accuracy from the *Error* grid, only non-empty cells in *Error* are used. In our framework we defined two accuracy error functions: N-th element error and the root mean square error (RMSE). The N-th element error allows us to report on the error given a certain percentage of total points. For instance, if the median error (threshold of 50%) is our measure for accuracy, the *N/2*-th *Error* would be reported. *N* is the total number of overlapping valid truth and input grid cells. Similarly reasoning can be done for the root mean square error (RMSE) also estimated in the benchmark software. In this case, the *Error* measure in algorithm 1 is replaced by the corresponding RMSE formula: $\sqrt{(In_{ij} - Truth_{ij})^2 / N}$.

Algorithm 1 Accuracy error metric

procedure ACCURACYERROR($Truth, In$)
  $Error \leftarrow$ Empty grid same size as $Truth$
  for $Truth_{i,j} \in Truth$ **do**
    if $Truth_{i,j}$ is valid & $In_{i,j}$ is not empty **then**
      $Error_{i,j} = |In_{i,j} - Truth_{i,j}|$
  return n_{th} element($Error_{i,j}$)

C. Completeness

Completeness is the fraction of valid truth cells with a corresponding non-empty input value within a threshold of the truth value for that cell in terms of Z error. Empty input grid cells are counted as incomplete as the MVS algorithm fails at placing a 3D point in that cell within the threshold. The detailed algorithm for completeness is shown in Algorithm 2.

Algorithm 2 completeness metric

procedure COMPLETENESS($Truth, In, Threshold, m, n$)
  for $Truth_{i,j} \in Truth$ **do**
    if $Truth_{i,j}$ is valid **then**
      $Total = Total + 1$
      if $In_{i,j}$ is not empty **then**
        if $|In_{i,j} - Truth_{i,j}| < Threshold$ **then**
          $Complete = Complete + 1$
  return $\frac{Complete}{Total}$

D. Point Cloud Registration

Registration handles aligning the truth and input data and removes any translational shift between them. The algorithm for removing translational shift searches through a set of possible translations and selects the position that minimizes the median error of the input. For performance reasons, registration does not search through all possible translations. Instead, it approximates by searching through a series of progressively finer search grids.

Our registration algorithm aims at minimizing the median error between the truth and the input by performing a full search in the truth reference grid. It does so at several resolution levels or *spacing*, starting from a coarse 3-meter spaced grid to half the spatial resolution of the truth grid. At each level it finds the translational shift that minimizes (median) error. Subsequent levels refine the results to find the optimal registration between truth and input grids. Algorithm 3 shows the registration process. The Refine function re-centers the search space around S_{best} and reduces the spacing between the possible shifts contained in *S*. *CreateGrid* is the cloud-to-grid conversion process described in section VIII.A with the translation *Shift* applied to all points in In_{cloud}. $Spacing_{desired}$ is the final desired spacing for the search grid. We set $Spacing_{desired}$ to half of the spatial resolution of the truth grid. The initial search space *S* is set to a *3* meter spaced square grid of possible translations covering a maximum of ± 27 meters.

Algorithm 3 Translation Registration

procedure FINDSHIFT($Truth, In_{cloud}, S$)
  for $Shift \in S$ **do**
    $In_{grid} \leftarrow$ CreateGrid($In_{cloud}, Shift$)
    $Error_{this} \leftarrow$ MedianError($Truth, In_{grid}$)
    if $Error_{min} > Error_{this}$ **then**
      $Error_{min} = Error_{this}$
      $Shift_{best} = Shift$
  if Spacing(S) $> Spacing_{desired}$ **then**
    $S_r \leftarrow$ Refine(S, S_{best})
    return FindShift($Truth, In_{cloud}, S_r$)
  else
    return $Shift_{best}$

IX. METRIC ANALYSIS

Metric analysis results are shown in Table 1 for point clouds produced during the IARPA Multi-View Stereo 3D Mapping Challenge Explorer contest on TopCoder. Results for the JHU/APL example implementation provided in the benchmark dataset are also shown for comparison. Solutions are shown in descending order of completeness (the fraction of points with Z error less than 1 meter). Ground truth lidar and solutions for this dataset are shown in Figure 8. Initial results are promising and demonstrate characteristic differences in the various approaches. While the error metric values are somewhat inflated here due to unfiltered seasonal features (e.g., foliage) and other temporally varying features, there appears to be significant room for improvement. Additional challenge areas are provided in the benchmark data that include a broad range of building heights and spacing as well as other variation in scene content.

Table 1 Scores Sorted by Completeness Metric

Solver	Completeness	Median (m)	RMS (m)
CarloDef	79.0%	0.260	2.615
KevinLaTourette	74.7%	0.278	2.727
SDRDIS	73.8%	0.180	3.401
JHU/APL Example	72.3%	0.322	2.720

X. CONCLUSION

The multiple view stereo benchmark for satellite imagery provides imagery, ground truth lidar data, and software to support research, development, and metric analysis. Initial analysis results using this dataset are demonstrated in this paper. Downloads and additional information regarding open source multiple view stereo solutions are available at http://www.jhuapl.edu/satellite-benchmark.html.

ACKNOWLEDGEMENT

WorldView-3 imagery was provided courtesy of Digital Globe. This work was sponsored by IARPA under contract 2012-12050800010. The authors would like to thank Booz Allen Hamilton, TopCoder, and everyone who participated in the IARPA Multi-View Stereo 3D Mapping Challenge. Due to scheduling constraints, only a few initial results from their work are shown in this paper.

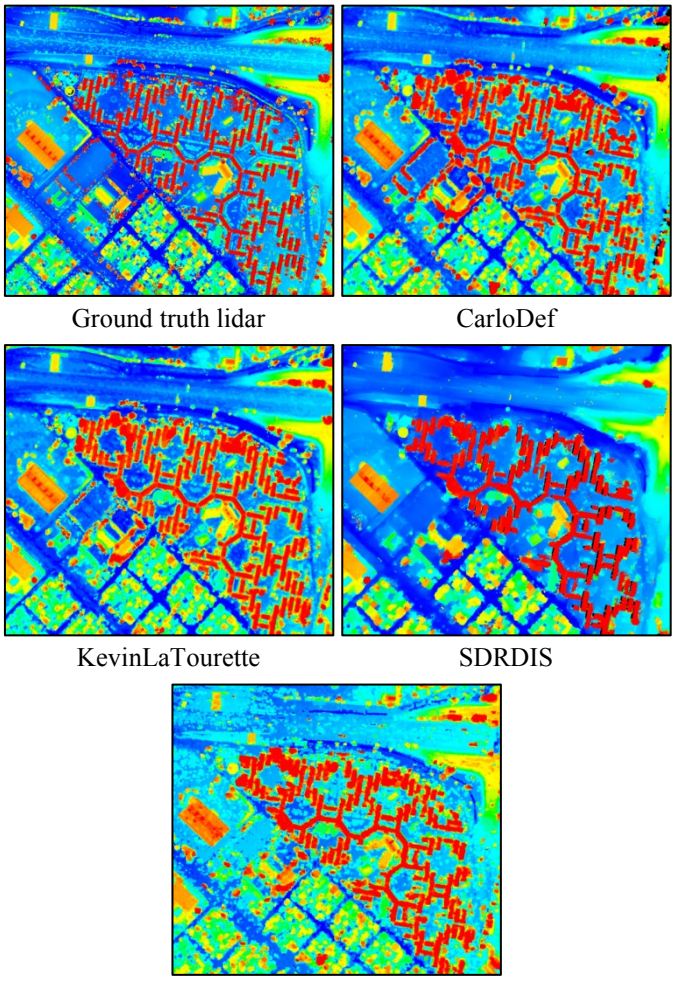

Figure 8. Explorer Dataset Point Clouds.

REFERENCES

[1] https://www.planet.com/products/open-california

[2] https://aws.amazon.com/public-data-sets/landsat/

[3] https://aws.amazon.com/public-data-sets/spacenet/

[4] S. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. "A comparison and evaluation of multi-view stereo reconstruction algorithms". In IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'2006), volume 1, pages 519–526, June 2006.

[5] R Jensen, A Dahl, G Vogiatzis, E Tola, H Aanaes, ," Large Scale Multi-view Stereopsis Evaluation"IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014, 406-413

[6] C. Strecha, W. von Hansen, L. J. V. Gool, P. Fua, and U. Thoennessen. "On benchmarking camera calibration and multi-view stereo for high resolution imagery". In Conference on Computer Vision and Pattern Recognition (CVPR), 2008.

[7] A. Geiger, P. Lenz, and R. Urtasun. "Are we ready for autonomous driving? the kitti vision benchmark suite". Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2012.

[8] http://vision.middlebury.edu/mview/

[9] http://www.cvlibs.net/datasets/kitti/

[10] https://www.iarpa.gov/index.php/working-with-iarpa/prize-challenges

[11] V. Tao and Y. Hu. "3d reconstruction methods based on the rational function model". ASPRS Photogrammetic Engineering and Remote Sensing Journal, 68(7):705–714, 2002.

[12] Z. Moratto, M. Broxton, R. Beyer, M. Lundy, and K. Husmann. "Ames stereo pipeline, nasa's open source automated stereogrammetry software". In Proceedings of the Lunar and Planetary Science Conference, page 2364, 2010.

[13] M. Broxton and L. Edwards. "Ames stereo pipeline: Automated 3d surface reconstruction from orbital imagery". In Proceedings of the Lunar and Planetary Science Conference, page 2419, 2008.

[14] E. Zheng, K. Wang, E. Dunn, and J. M. Frahm. "Minimal solvers for 3d geometry from satellite imagery". In 2015 IEEE International Conference on Computer Vision (ICCV), pages 738–746, Dec 2015. 2

[15] G. Kuschk. "Large Scale Urban Reconstruction from Remote Sensing Imagery". ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, (1):139–146, 2013.

[16] D. Crispell, J. L. Mundy, and G. Taubin. "A variable resolution probabilistic three-dimensional model for change detection". IEEE Transactions on Geoscience and Remote Sensing, 2011.

[17] T. Pollard, I. Eden, J. Mundy, and D. Cooper. "A volumetric approach to change detection in satellite images". ASPRS Photogrammetic Engineering and Remote Sensing Journal, 76(7):817–831, 2010.

[18] O. Ozcanli, Y. Dong, J. Mundy, H. Webb, R. Hammond, and V. Tom. "A comparison of stereo and multiview 3-d reconstruction using cross-sensor satellite imagery". In Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), pages 17–25, 2015.

[19] H. Aanæs, A. L. Dahl, and V. Perfanov. "A ground truth data set for two view image matching". Technical report, DTU Informatics, Technical University of Denmark, 2010.

[20] A. Ulusoy, M. Black, and A. Geiger. "Patches, Planes and Probabilities: A Non-local Prior for Volumetric 3D Reconstruction". Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[21] K. Wilson and N. Snavely. "Robust global translations with 1 dsfm". In Proceedings of the European Conference on Computer Vision, 2014.

[22] N. Haala. "Benchmark on image matching". Technical report, Institute for Photogrammetry, University of Stuttgart, 2014.

[23] M. Mandani. "Real-time sensor independent positioning by rational functions". In Proceedings of the ISPRS Workshop on Direct vs. Indirect Methods of Sensor Orientations, pages 64–75, 1999.

[24] G. Dial and J. Grodecki. "RPC replacement camera model". In Proceedings of the ASPRS, 2005.

[25] "The compendium of controlled extensions for the national imagery transmission format (nitf)". Tech.report, National Imagery Agency2000.

[26] M. A. Fischler and R. C. Bolles. "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography". Commun. ACM, 24(6):381–395, June 1981.

[27] C. Liu, J. Yuen, A. Torralba, J. Sivic, and W. T. Freeman. "Sift flow: Dense correspondence across different scenes". Proceedings of the 10th European Conference on Computer Vision: Part III, pages 28–42, 2008.

[28] Y. Li, J. Yuen, A. Torralba, J. Sivic, and W. T. Freeman. "Exploiting reflection change for automatic reflection removal". In Proceedings of the IEEE International Conference on Computer Vision (ICCV), pages 2432–2439, 2013.

[29] O. C. Ozcanli, Y. Dong, J. L. Mundy, H. Webb, R. Hammoud, and V. Tom. “Automatic geolocation correction of satellite imagery”. International Journal of Computer Vision, 116(3):263–277, 2016.

[30] M. Kuester, "Radiometric use of WorldView-3 Imagery", February 2016, https://www.digitalglobe.com/resources/technical-information

[31] R. I. Hartley and A. Zisserman. “Multiple View Geometry” in Computer Vision. Cambridge University Press, ISBN:0521540518, sec. ed., 2004.

[32] Carlo de Franchis, Enric Meinhardt-Llopis, Julien Michel, Jean-Michel Morel, and Gabriele Facciolo, "An automatic and modular stereo pipeline for pushbroom images",ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume II-3, 2014.

[33] Yasutaka Furukawa, Brian Curless, Steven M. Seitz, and Richard Szeliski. Towards Internet-scale multiview stereo. In IEEE Conference on Computer Vision and Pattern Recognition, 2010.

[34] Brian Curless and Marc Levoy. A volumetric method for building complex models from range images. In ACM SIGGRAPH, 1996.

[35] C. Baillard, C. Schmid, A. Zisserman, and A. W. Fitzgibbon. Automatic line matching and 3D reconstruction of buildings from multiple views. In ISPRS Conference on Automatic Extraction of GIS Objects from Digital Imagery, pages 69–80, 1999.

[36] Sameer Agarwal, Noah Snavely, Ian Simon, Steven M. Seitz, and Richard Szeliski. Building Rome in a day. In IEEE International Conference on Computer Vision, 2009.

[37] Y. Furukawa, and C. Hernandez, “Multi-View Stereo: A Tutorial”, Foundations and Trends in Computer Graphics and Vision Series, Now Publishers Incorporated, 2015.

[38] S. Fuhrmann, F. Langguth, and M. Goesele, “MVE-A Multi-View Reconstruction Environment”, Proceedings of the Eurographics Workshop on Graphics and Cultural Heritage, Darmstadt, Germany 2014.

[39] E. Tola, C. Strecha and P. Fua, “Efficient large-scale multi-view stereo for ultra high-resolution image sets”, Machine Vision and Applications, May 2011.

[40] A. Ulusoy, M. Black, and A. Geiger, “Patches, Planes and Probabilities: A Non-local prior for volumetric 3D reconstruction”, Proceedings of IEEE on Computer Vision and Pattern Recognition (CVPR), Las Vegas, USA, 2016.

[41] http://www2.isprs.org/commissions/comm1/wg2/benchmark.html