

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/329751459>

# A Blender plug-in for comparing Structure from Motion pipelines

Conference Paper · September 2018

DOI: 10.1109/ICCE-Berlin.2018.8576196

---

CITATIONS

6

READS

919

4 authors:



Davide Marelli

Università degli Studi di Milano-Bicocca

8 PUBLICATIONS 102 CITATIONS

[SEE PROFILE](#)



Simone Bianco

Università degli Studi di Milano-Bicocca

148 PUBLICATIONS 3,020 CITATIONS

[SEE PROFILE](#)



Luigi Celona

Università degli Studi di Milano-Bicocca

31 PUBLICATIONS 845 CITATIONS

[SEE PROFILE](#)



Gianluigi Ciocca

Università degli Studi di Milano-Bicocca

141 PUBLICATIONS 2,784 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Visual quality assessment [View project](#)



Single image dehazing [View project](#)

# A Blender plug-in for comparing Structure from Motion pipelines

Davide Marelli, Simone Bianco, Luigi Celona, Gianluigi Ciocca

University of Milano-Bicocca

DISCo - Department of Computer Science, Systems and Communications

Viale Sarca 336, 20126, Milan, Italy

d.marelli8@campus.unimib.it, bianco@disco.unimib.it, luigi.celona@disco.unimib.it, ciocca@disco.unimib.it

**Abstract**—Structure from Motion (SfM) is a pipeline that allows three-dimensional reconstruction starting from a collection of images. A typical SfM pipeline comprises different processing steps each of which tackle a different problem in the reconstruction pipeline. Each step can exploit different algorithms to solve the problem at hand and thus many different SfM pipelines can be built. There are many SfM pipelines available in the literature. How to choose the best among them? We present a Blender plug-in that provides an easy to use tool to compare them under different conditions using both real and synthetic datasets.

**Index Terms**—Structure from Motion, 3D reconstruction, Blender.

## I. INTRODUCTION

Three-dimensional reconstruction is the process that allows to capture the geometry and appearance of an object or an entire scene. In the last years, interests has developed around the use of 3D reconstruction for gaming, virtual and augmented reality. Those techniques have been used to realize video game assets [1], [2], virtual tours [3] as well as mobile 3D reconstruction apps [4], [5]. Some other areas in which 3D reconstruction can be used are CAD (Computer Aided Design) software [6], computer graphics and animation [7], [8], diagnostic imaging [9], virtual reality [10], etc...

Over the years, various techniques and algorithms for 3D reconstruction have been developed to meet different needs in different fields of application ranging from active methods that require the use of special equipment to capture geometry information (i.e. laser scanners, structured lights, microwaves, ultrasound, etc...) to passive methods that are based on optical imaging techniques only. The latter techniques do not require special devices or equipment and thus are easily applicable in different contexts. Among the passive techniques for 3D reconstruction there is the Structure Structure from Motion (SfM) pipeline [11]–[15]. Given a set of images acquired from different observation points, it recovers the pose of the camera for each input image and a three-dimensional reconstruction of the scene in form of a sparse point cloud. After this first sparse reconstruction, it is possible to run a dense reconstruction phase using Multi-View Stereo (MVS) [16].

As it can be seen from Figure 1, a typical SfM pipeline comprises different processing steps each of which tackle a different problem in the reconstruction pipeline. Each step can

exploit different algorithms to solve the problem at hand and thus many different SfM pipelines can be built. There are many SfM pipelines available in the literature. How to choose the best among them?

In this paper, we compare different state-of-the-art SfM pipelines in terms of their ability to reconstruct different scenes. The comparison is carried out by evaluating the reconstruction error of each pipeline on an evaluation dataset. The dataset is composed of real objects whose ground truth has been acquired with high-end devices. Having real scenes as reference models is no trivial thus we have developed a plug-in for Blender to create an evaluation dataset starting from synthetic 3D scenes. This allows us to rapidly and efficiently extend the existing dataset and stress the pipelines under different conditions. In this way it is possible to evaluate solutions for 3D reconstruction on consumer electronic imaging devices (eg. smartphones, tablets, etc.), which potentially allow users to select the best pipeline to generate their own 3D contents.

## II. INCREMENTAL SFM PIPELINE

In this paper we focus our attention on the incremental SfM pipeline. This approach is the most common strategy to achieve 3D reconstruction from unordered image collections without prior information on camera pose (position and orientation). Recent developments have led to excellent results in reconstruction accuracy. The incremental reconstruction pipeline, shown in Figure 1, is a sequential pipeline that consists of a first phase of correspondences search between images and a second phase of iterative incremental reconstruction that starts from an initial two-view reconstruction that is progressively extended by adding new images.

### A. SfM building blocks

**Feature Extraction** - for each image in input to the pipeline a collection of local features is created to describe its key points. For this step different algorithms can be used, such as: SIFT, SURF and ORB.

**Feature Matching** - key points and features obtained are then used to determine which images portray common parts of the scene calculating a measure distance between features. The most used algorithms use one of the following solutions:

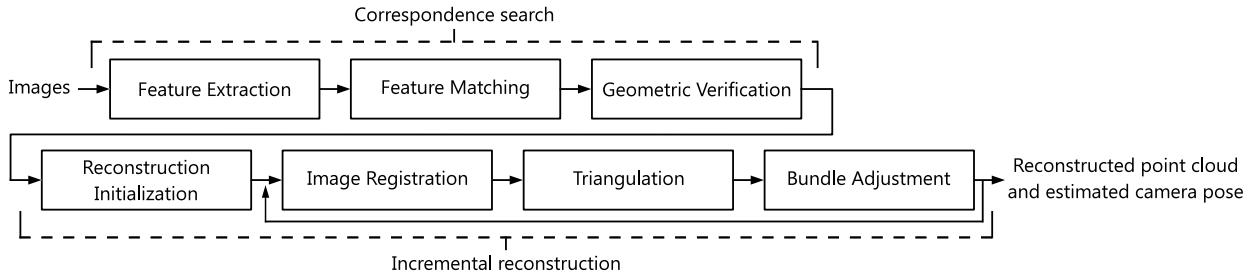


Fig. 1. Incremental Structure from Motion (SfM) pipeline

L2-norm, Sum of Squared Differences, Sum of Absolute Differences, Hamming distance.

*Geometric Verification* - checks if the matches found are also corresponding 3D points searching a valid geometric transformation between image pairs. Since results from previous step are often contaminated by outliers, robust estimation techniques such as RANSAC [17], PROSAC [18], ARRSAC [19], EVSAC [20] or LMed [21] are used.

*Reconstruction Initialization* - a pair of strongly geometrically verified images is chosen to initialize a two-view reconstruction. This is an important phase because a bad initialization leads to a bad reconstruction of the 3D model.

*Image Registration* - is the first incremental step, a new image is considered and the pose of the associated camera is estimated by solving the Perspective-n-Point (PnP) problem. Various algorithms can be used such as: P3P [22], EPnP [23], Direct Linear Transformation (DLT) [24], Normalized 8-point algorithm [24], Direct Least-Squares (DLS) [25], P4P [26].

*Triangulation* - the new registered image can observe new points in the scene; those can be triangulated if are observed by at least another previously registered image. The process is based on an epipolar constraint, however because of the inaccuracies in previous steps the constraint is relaxed using: 2-View triangulation [27], Midpoint triangulation [28], Linear triangulation [24] or N-View triangulation [24].

*Bundle Adjustment* - the purpose of this phase [29] is to prevent inaccuracies in estimated camera pose to propagate in the triangulation of 3D points and vice versa. A commonly used algorithm is Levenberg-Marquardt, also known as Damped Least-Squares. This phase has an high computational cost and must be executed for each image.

### B. Image acquisition

In order to make a reconstruction, a set of images of the object or scene to rebuild is needed. For an optimal reconstruction the dataset should respect the following guidelines:

- The object to be reconstructed must not have a too uniform geometry and must have a varied texture. An object with uniform geometry and repeated or monochromatic texture makes it difficult to estimate the camera pose.
- The set must be composed of a number of images sufficient to cover, at least in pairs, the entire surface of the object to be rebuilt.

- The quality of the reconstruction also depends on the quality of the images. A set of images with good resolution and level of detail should lead to a good reconstruction.
- The intrinsic parameters of the camera must be known for each image. In particular, the pipeline makes use of focal length and sensor size to estimate the camera pose and to generate the sparse point cloud.

If the set of images does not comply with one or more of the above guidelines, the results of pipeline execution may vary from a low quality up to an incorrect reconstruction.

### III. EVALUATION PLUG-IN

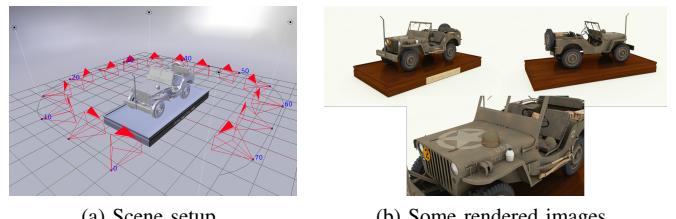


Fig. 2. Example of synthetic dataset generation.

Generation of datasets encounters limitations due to the acquisition equipment or the scene to be captured itself. To overcome these problems it is possible to use virtual 3D models to generate datasets with good image quality, intrinsic parameters for each image and optimal 3D model ground truth.

Synthetic dataset creation, pipeline execution and results evaluation involve many steps and various algorithms. To help the user in the process we created a plug-in for Blender that allows synthetic datasets generation and SfM reconstructions evaluation. Such tool adds a simple panel in Blender's user interface that makes possible to:

- import the main object of the reconstruction and setup a scene with lights for illumination and uniform background walls. Also, the parameters for the path tracing rendering engine are set.
- add a camera and setup its intrinsic calibration parameters. Animate the camera using circular rotations around the object to observe the scene from different view points.
- render the set of images and add EXIF metadata of intrinsic camera parameters used by SfM pipelines.



Fig. 3. 3D models used for synthetic dataset generation (a-e) and ground truth of Ignatius dataset (f).

- eventually, geometry ground truth can be exported. This is not necessary if next steps are done using this plug-in as the current scene will be used as ground truth.
- run the SfM pipelines listed in Section IV-A.
- import the reconstructed point cloud from SfM output and allow the user to manually eliminate parts that do not belong to the main object of the reconstruction.
- align the reconstructed point cloud to the ground truth using the Iterative Closest Point algorithm (ICP).
- evaluate the reconstructed cloud computing the distance between the cloud and the ground truth and generating statistic information like min, max and average distance values and also reconstructed point count.

The dataset generation process could be used to realize set of images of scene with various scale and likely include many objects, for those reason the plug-in divides the process in different steps, in this way it is possible for the user to adapt the result obtained after each step to specific needs. For example, it is possible to change the default camera intrinsic parameters, the scene illumination, animate the camera with different paths than the defaults and so on. An example of scene setup and rendered images is visible Figure 2.

#### IV. SfM PIPELINES COMPARISON AND RESULTS

##### A. Evaluated SfM implementations

We have considered for evaluation different SfM pipelines, specifically:

- COLMAP [12] - an open-source implementation with an intuitive graphical interface. The main objective of its creators is to provide a general-purpose solution usable to reconstruct any scene introducing also enhancements in robustness, accuracy and scalability.
- Theia [30] - an open-source SfM library that includes many algorithms commonly used for 3D reconstruction. In particular many options are provided for the camera pose estimation phase. It is also possible to extend the library with new algorithms using its software interfaces.
- OpenMVG [31] - an open-source library to solve Multiple View Geometry problems that includes an implementation of the SfM pipeline with different options for 3D reconstruction. It is also possible to use geographic data and GPS coordinates for the pose estimation phase.
- VisualSfM [14] - one of the reference implementations, comes with an intuitive graphical user interface. Compared to other solutions, this one is less flexible because includes only one set of algorithms.

These pipelines were chosen because each one is a reference implementation. In particular VisualSfM and COLMAP represent two remarkable developments of the incremental SfM pipeline with improvements in accuracy and performance compared to previous state-of-the-art implementations. Theia and OpenMVG are instead two ready to use SfM and multi-view geometry libraries that implement reconstruction algorithms and allow to build SfM pipelines that meet specific needs.

##### B. Evaluation datasets

Five synthetic datasets have been generated (Figure 3). All the images of each dataset have been acquired at resolution  $1920 \times 1080\text{px}$  using a virtual camera with a 35mm focal length and  $18 \times 32\text{mm}$  sensor. In addition to our synthetic datasets the real dataset Ignatius from the “Tanks and Temples” collection [32] is also used whose images have been acquired at a resolution of  $1920 \times 1080\text{px}$ . The final dataset, that can be downloaded from <http://www.ivl.disco.unimib.it/?p=1067>, is composed of:

- Statue [33] - set of 121 images about a statue
- Empire Vase [34] - set of 86 images about a vase
- Bicycle [35] - set of 86 images about a bicycle
- Hydrant [36] - set of 66 images about an hydrant
- Jeep [37] - set of 141 images about a miniature jeep
- Ignatius [32] - set of 263 images about a real statue

##### C. Evaluation method

The sparse cloud generated by SfM is compared to the ground truth calculating the distance between the two. Since the reconstruction and the ground truth use different reference coordinate systems, it is necessary to find the correct alignment between the two. Translation, rotation, and scale factors to align the two can be defined using a rigid transformation matrix. Once the reconstruction is aligned to the ground truth it is possible to proceed with the evaluation of the reconstructed point cloud.

If the ground truth is available as a dense point cloud, for each 3D point of the cloud to be compared, the nearest point is searched in the reference cloud calculating the Euclidean distance. If the ground truth is available as a mesh, the distance is calculated between a reconstructed point and the nearest point on the triangles of the mesh [38]. Once the distance values are obtained for all cloud’s points, the mean value and standard deviation can be calculated.

In both cases it is necessary that the reconstructed cloud contains only points relative to objects that are included in the ground truth model used for comparison. Usually the ground

truth includes only the main object of the reconstruction, ignoring the other elements visible in the dataset's images. If the reconstruction includes parts of the scene that do not belong to the ground truth, the distance calculation will be distorted. To overcome this problem, it is possible to cut out the reconstructed cloud manually eliminating the excess parts before evaluating the distance. If this is not possible (mainly because the separation between the objects of interest and those not relevant is not simply identifiable), then the same result can be achieved by discarding points that are beyond than a specified maximum limit.

#### D. Results

A synthesis of obtained results is reported in Table I; for each reconstruction are reported the mean distance of the reconstructed point cloud from the ground truth and the number of points that compose such cloud. Figure 4 shows some examples of the reconstructed sparse point cloud generated by COLMAP.

TABLE I  
RECONSTRUCTION RESULTS

	Point cloud mean distance [m] (points no.)			
	COLMAP	OpenMVG	Theia	VisualSfM
Statue	0.034 (9k)	0.057 (4k)	0.020 (8k)	0.185 (6k)
E. Vase	0.005 (8k)	0.013 (2k)	0.002 (8k)	0.007 (5k)
Bicycle	0.042 (5k)	0.156 (7k)	0.027 (2k)	0.056 (4k)
Hydrant	0.206 (2k)	– (28)	0.045 (89)	0.029 (1k)
Jeep	0.053 (6k)	0.057 (4k)	0.012 (8k)	0.055 (5k)
Ignatius	0.009 (23k)	0.013 (12k)	0.023 (10k)	0.054 (14k)

SfM pipelines have generated sufficient information on all datasets except for the Hydrant one. That dataset has a low geometric complexity, an high level of symmetry and an almost uniform texture; for these reasons SfM pipelines were not able to find enough correspondence between images and thus cannot generate a good reconstruction. The worst reconstruction is that of OpenMVG where the limited number of points does not allow to evaluate the reconstructed point cloud.

The real dataset Ignatius has the highest number of reconstructed points, this is partly due to the presence of other elements in the scene that allow precise camera pose estimation and point triangulation. Also note that point count refers to only the ones that belong to the main object of the scene.

COLMAP and Theia are the pipelines that achieve better results on cloud distance and number of points. COLMAP is also the pipeline that reconstructs the highest number of points on critical datasets like Hydrant and Bicycle while maintaining acceptable values for the cloud distance. It also generates the most dense and accurate cloud on the real dataset Ignatius.

#### V. CONCLUSIONS

In this paper we analyzed the state-of-the-art incremental SfM pipelines showing that different algorithms and approaches can be used for each step of the reconstruction

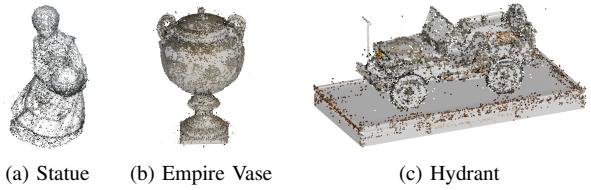


Fig. 4. Examples of reconstructed sparse point cloud.

process. We proposed a complete method that starting from synthetic dataset generation allows to overcome real datasets limitations, evaluate and compare the obtainable reconstruction from different SfM implementations testing theirs limits under different conditions.

Experimental results show that is possible to generate synthetic datasets from which SfM reconstruction can successfully run obtaining satisfactory results. This also allows to take the pipelines to their limits showing that critical conditions can affect negatively the reconstruction process. In experiments the incremental SfM implementations of COLMAP and Theia showed the best results on average.

We also created a software tool that allows in a single solution to run the whole process, from the dataset generation to the reconstruction evaluation.

Further work can be done to evaluate other aspects of the pipeline such as camera poses and reconstructed object coverage to identify missing parts. The evaluation method can also be extended to include the subsequent dense reconstruction phase. We have already done an investigation on such aspects and preliminary results can be found in [39].

#### REFERENCES

- [1] A. Poznanski, “Visual revolution of the vanishing of ethan carter,” <http://www.theastronauts.com/2014/03/visual-revolution-vanishing-ethan-carter/>, 2014, accessed 25-May-2018.
- [2] A. Hamilton and K. Brown, “Photogrammetry and star wars battlefield,” <https://www.ea.com/frostbite/news/photogrammetry-and-star-wars-battlefront>, 2016, accessed 25-May-2018.
- [3] O. Saurer, F. Fraundorfer, and M. Pollefeys, “Omnitour: Semi-automatic generation of interactive virtual tours from omnidirectional video,” in *Proc. 3DPVT2010 (Int. Symp. on 3D Data Processing, Visualization and Transmission)*, 2010.
- [4] O. Muratov, Y. Slyntko, V. Chernov, M. Lyubimtseva, A. Shamsurov, and V. Bucha, “3dcapture: 3d reconstruction for a smartphone,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2016, pp. 75–82.
- [5] SmartMobileVision, “Scann3d,” <http://www.smartmobilevision.com>, accessed 25-May-2018.
- [6] F. Bernardini, C. L. Bajaj, J. Chen, and D. R. Schikore, “Automatic reconstruction of 3d cad models from digital scans,” *International Journal of Computational Geometry & Applications*, vol. 9, no. 04n05, pp. 327–369, 1999.
- [7] S. Izadi, D. Kim, O. Hilliges, D. Molynieux, R. Newcombe, P. Kohli, J. Shotton, S. Hodges, D. Freeman, A. Davison *et al.*, “Kinectfusion: real-time 3d reconstruction and interaction using a moving depth camera,” in *Proceedings of the 24th annual ACM symposium on User interface software and technology*. ACM, 2011, pp. 559–568.
- [8] J. Starck and A. Hilton, “Surface capture for performance-based animation,” *IEEE computer graphics and applications*, vol. 27, no. 3, 2007.
- [9] I. Carlborn, D. Terzopoulos, and K. M. Harris, “Computer-assisted registration, segmentation, and 3d reconstruction from images of neuronal tissue sections,” *IEEE Transactions on medical imaging*, vol. 13, no. 2, pp. 351–362, 1994.

- [10] Z. Noh and M. S. Sunar, *A Review on 3D Reconstruction for Augmented Reality in Virtual Heritage System*, 2009.
- [11] O. Özyeşil, V. Voroninski, R. Basri, and A. Singer, “A survey of structure from motion,” *Acta Numerica*, vol. 26, pp. 305–364, 2017.
- [12] J. L. Schönberger and J.-M. Frahm, “Structure-from-motion revisited,” in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [13] J. Ko and Y.-S. Ho, “3d point cloud generation using structure from motion with multiple view images,” in *The Korean Institute of Smart Media Fall Conference*, 2016, pp. 91–92.
- [14] C. Wu, “Towards linear-time incremental structure from motion,” in *3D Vision-3DV 2013, 2013 International conference on*. IEEE, 2013, pp. 127–134.
- [15] N. Snavely, S. M. Seitz, and R. Szeliski, “Photo tourism: exploring photo collections in 3d,” in *ACM transactions on graphics (TOG)*, vol. 25, no. 3. ACM, 2006, pp. 835–846.
- [16] Y. Furukawa, C. Hernández *et al.*, “Multi-view stereo: A tutorial,” *Foundations and Trends® in Computer Graphics and Vision*, vol. 9, no. 1-2, pp. 1–148, 2015.
- [17] M. A. Fischler and R. C. Bolles, “Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography,” in *Readings in computer vision*. Elsevier, 1987, pp. 726–740.
- [18] O. Chum and J. Matas, “Matching with prosac-progressive sample consensus,” in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 220–226.
- [19] R. Raguram, J.-M. Frahm, and M. Pollefeys, “A comparative analysis of ransac techniques leading to adaptive real-time random sample consensus,” in *European Conference on Computer Vision*. Springer, 2008, pp. 500–513.
- [20] V. Fragoso, P. Sen, S. Rodriguez, and M. Turk, “Evsac: accelerating hypotheses generation by modeling matching scores with extreme value theory,” in *Computer Vision (ICCV), 2013 IEEE International Conference on*. IEEE, 2013, pp. 2472–2479.
- [21] P. J. Rousseeuw, “Least median of squares regression,” *Journal of the American Statistical Association*, vol. 79, no. 388, pp. 871–880, 1984.
- [22] X.-S. Gao, X.-R. Hou, J. Tang, and H.-F. Cheng, “Complete solution classification for the perspective-three-point problem,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 25, no. 8, pp. 930–943, 2003.
- [23] V. Lepetit, F. Moreno-Noguer, and P. Fua, “Epnp: An accurate  $O(n)$  solution to the pnp problem,” *International journal of computer vision*, vol. 81, no. 2, p. 155, 2009.
- [24] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.
- [25] J. A. Hesch and S. I. Roumeliotis, “A direct least-squares (dls) method for pnp,” in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 383–390.
- [26] M. Bujnak, Z. Kukelova, and T. Pajdla, “A general solution to the p4p problem for camera with unknown focal length,” in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.
- [27] P. Lindstrom, “Triangulation made easy,” in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 1554–1561.
- [28] R. I. Hartley and P. Sturm, “Triangulation,” *Computer vision and image understanding*, vol. 68, no. 2, pp. 146–157, 1997.
- [29] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon, “Bundle adjustment - a modern synthesis,” in *International workshop on vision algorithms*. Springer, 1999, pp. 298–372.
- [30] C. Sweeney, “Theia multiview geometry library: Tutorial & reference,” <http://theia-sfm.org>.
- [31] P. Moulon, P. Monasse, R. Marlet, and Others, “Openmvg. an open multiple view geometry library.” <https://github.com/openMVG/openMVG>.
- [32] A. Knapsitsch, J. Park, Q.-Y. Zhou, and V. Koltun, “Tanks and temples: Benchmarking large-scale scene reconstruction,” *ACM Transactions on Graphics*, vol. 36, no. 4, 2017.
- [33] Free 3D user dgemmell1960, “Statue,” <https://free3d.com/3d-model/statue-92429.html>, 2017, accessed 25-May-2018.
- [34] Blend Swap user geoffreymarchal, “Empire vase,” <https://www.blendswap.com/blends/view/90518>, 2017, accessed 25-May-2018.
- [35] Blend Swap user milkyduhan, “Bicycle textured,” <https://www.blendswap.com/blends/view/67563>, 2013, accessed 25-May-2018.
- [36] Blend Swap user b82160, “High poly hydrant,” <https://www.blendswap.com/blends/view/87541>, 2017, accessed 25-May-2018.
- [37] Blend Swap user BMF, “Willys jeep circa 1944,” <https://www.blendswap.com/blends/view/82687>, 2016, accessed 25-May-2018.
- [38] D. Eberly, “Distance between point and triangle in 3d,” *Magic Software*, <http://www.magic-software.com/Documentation/pt3tri3.pdf>, 1999.
- [39] S. Bianco, G. Ciocca, and D. Marelli, “Evaluating the performance of structure from motion pipelines,” *Journal of Imaging, Special Issue on Image Enhancement, Modeling and Visualization*, 2018.