Development of a Remote Rock Fragmentation Size Distribution Measurement System for Surface Mines Using 3D Photogrammetry

Z. P. L. Tungol[a]*, Y. Kawamura[a], I. Kitahara[b], H. Jang[c]

[a] Akita University, Akita, Japan

[b] University of Tsukuba, Tsukuba, Japan

[c] Curtin University, Kalgoorlie, Australia

ABSTRACT: One of the factors that can affect the efficiency of a mining operation is the fragmentation size distribution of blasted rock. A consistent fragmentation size allows the company to streamline its process, and more importantly, minimize costs. In order to maintain this fragmentation size, monitoring must be done regularly so that adjustments can be made. Traditional methods such as manual sieving and visual estimation are have been used for this purpose, but limitations on sampling procedure and bias make these methods relatively inefficient. One of the solutions that were developed was to use digital image-based particle size analysis. The study proposes a cloud-based 3D photogrammetry rock fragmentation size distribution system that will make use of multiple images to create 3D models that can then be analyzed and segmented to provide a fragmentation size distribution. Several pictures of a muckpile using a smartphone are taken from an angle and compiled into a dataset. This is used as input for a Structure-from-Motion algorithm, which can create a 3D point cloud from the image data. This point cloud is then subjected to clustering so that the individual fragments can be represented and their dimensions could be measured. Finally, from these dimensions, a fragmentation size distribution can be created. As the system requires a large amount of computing power, it can be implemented in a remote server so that it can be accessible in the field. This system could provide surface mine operators an easy way to estimate size distribution using only a smartphone.

Keywords: Fragmentation size, 3D Photogrammetry, Structure from Motion

1 INTRODUCTION

In any mining operation, there are a number of factors that can affect the efficiency of the day-to-day resource extraction process across all of its stages from mine (drilling, blasting, haulage) to mill (mineral processing). As such, it is up to the management, when possible, to monitor these various factors and act accordingly by making modifications to mine planning as well as tweaking its execution. One of such factors, specifically in mining operations that employ explosives and mineral processing, is the fragmentation size distribution of rock after it has been blasted. The mine and the mill usually have an agreed-upon average fragmentation size from which various elements of the operation will be based on, such as drilling equipment, explosives, and mineral processing machinery. The importance of this specification is due to how it generally affects mining and milling costs differently. When a relatively high fragmentation size is set, the cost of drilling and blasting is proportionately lower (Afum and Temeng 2014), but will require more mineral processing in order to achieve the final product. Inversely, a relatively lower fragmentation size means less expense is needed for milling, but the cost of drilling and blasting would increase (BME South Africa 2016). Considering these effects, a middle ground must be reached in order to minimize the total cost of the operation, as illustrated by a theoretical graph in Figure 1.

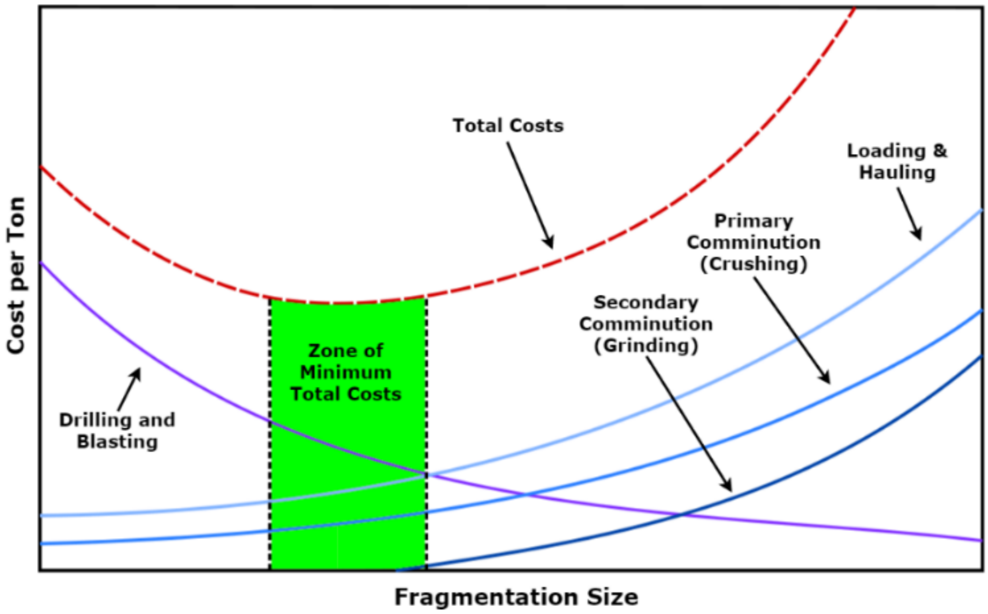

Figure 1. Relationship between fragmentation size and operational costs in mining

In order to maintain this aforementioned fragmentation size, monitoring must be done on a regular basis so that changes in the drilling and blasting process can be made. However, due to the heterogeneous nature of the rock strata, as well as the sheer amount of material that is continuously mined, it can be difficult to monitor the size distribution during day-to-day operations. Traditionally, manual sieving, boulder counting and visual estimation were used for this purpose, but limitations on sampling and bias made these methods relatively inefficient (Nefis and Talhi 2016). This then presented the need for an accurate, quick and accessible method of determining the rock fragmentation size distribution without intensive sampling and laboratory testing. One of the solutions that were proposed is to use image-based particle size analysis software. Current methods such as WipFrag make use of a single image of a muck pile to determine fragmentation size distribution. In an attempt to develop a new method that has the potential to improve upon the accuracy of this particular method, the study then proposes a system that will make use of multiple images from different angles to create 3-dimensional models that can then be analyzed and segmented to provide a fragmentation size distribution.

2 SYSTEM OVERVIEW

The proposed system is divided into stages, utilizing multiple computational techniques in order to achieve its purpose. In a hypothetical application of the system, pictures of the muck pile from the products of blasting are taken. The images are then processed in a high-spec computer by a sequence of 3d imaging techniques that will ultimately output a scaled 3d model of the muck pile. A technique known as supervoxel clustering is then performed on the 3d model then undergoes supervoxel clustering in order to divide the individual fragments into segments whose dimensions have been calculated. The dimensional data can then be used in the computation of the fragment size distribution of the muck pile. Using this information, the blasting product can be judged if it is up to the expected specification. Adjustments are then made the blasting design such as the amount and type of explosive and blasting patterns in order to achieve the required distribution. As of the time of writing of this manuscript, the research is currently working on the Supervoxel Segmentation stage of the process.

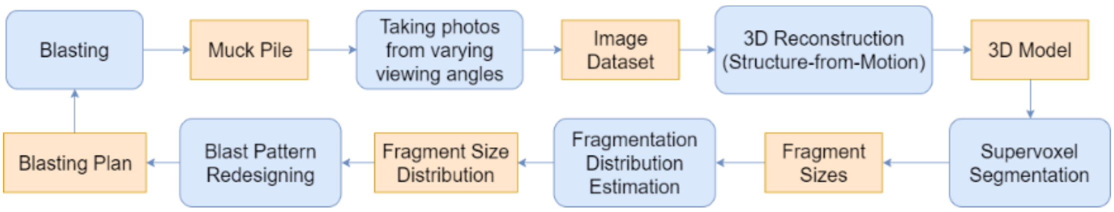

Figure 2. Proposed system workflow

3 STRUCTURE FROM MOTION – MULTI VIEW STEREO (SFM-MVS)

3.1 The General SfM-MVS Pipeline

It can be inferred that there is no single ‘correct’ workflow or process in the conversion of 2d images into models. However, there are key processes that are present in almost all applications of the method, as shown in Figure 3.

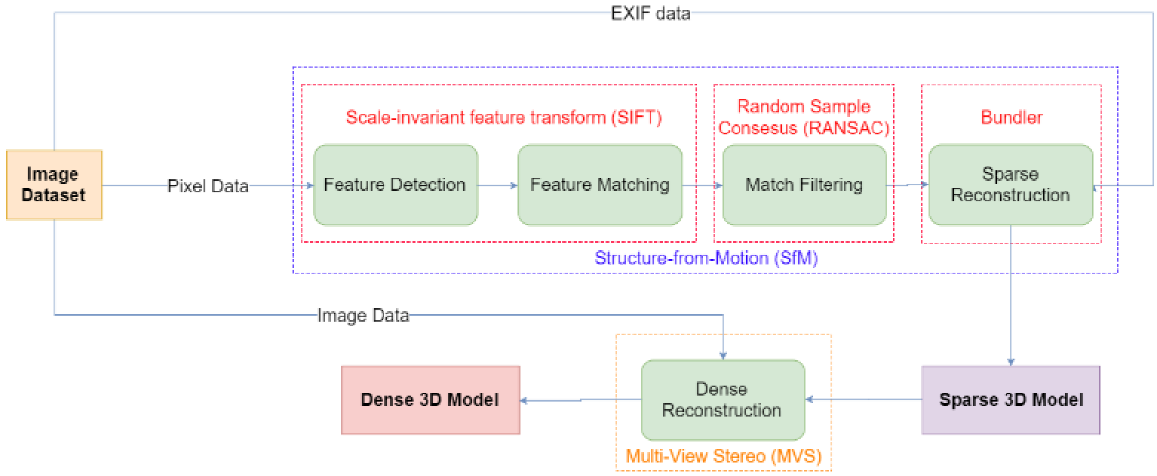

Figure 3. Typical SfM-MVS Pipeline

3.2 Feature Detection

The initial processing step after acquiring the images is feature detection, or extraction, where possible common features (or keypoints) in the individual images are identified (Westoby, et al. 2012). It is by these features that allow the different images in the dataset to be matched at the next stage. There are several techniques that have been developed for the solution of this step (Carrivick, Smith and Quincey 2016), but the most widely used amongst modern SfM applications is the scale-invariant feature transform (SIFT) object-recognition system which was developed by David G. Lowe in the University of British Columbia in 2004 (Lowe 2004). The system recognizes feature points in the image set which are uniform in scaling and rotation and relatively uniform to changes in lighting and 3-dimensional camera view angles. The number of keypoints that are extracted in an image relies heavily on the resolution and texture of the images themselves, with high-quality, original-resolution pictures returning the most results (Westoby, et al. 2012).

3.3 Feature Matching

The next step is to match the keypoints and identify the correspondences between them. Matches are found by identifying a keypoint's nearest neighbor in the database. The nearest neighbor is defined as the keypoint with the least Euclidean distance for its descriptor vector (Lowe 2004). It is also important to note at this point that not all keypoints are guaranteed to have a good match in the dataset. It is therefore necessary to discard these unmatched keypoints, making use of the ratio between the Euclidean distance of the nearest neighbor with that of the second nearest, at a certain minimum value as a criterion for discarding false keypoint matches. (Carrivick, Smith and Quincey 2016). The inherent complexity of the keypoint descriptors gives rise to the need of an efficient solution to the search process, as brute-force searching for nearest neighbors proves to be computationally difficult and time-consuming. Several solutions such as k-dimensional trees (k-d trees), best-bin first (BBF) and approximate nearest neighbor (ANN) searching are used to solve this problem of efficiency by partitioning the data into bins which are prioritized for match searching, decreasing the number of recursions needed to go through all the keypoints. (Carrivick, Smith and Quincey 2016)

3.4 Match Filtering

The third stage, also known as Geometric Verification, is done to further eliminate erroneous matches. Since the matching done by SIFT is solely based on appearance, it cannot be guaranteed that the matched keypoints refer to the same point in an image (e.g. images with symmetrical or similar features) (Schoenberger and Frahm 2016). SfM then needs to verify matches by mapping keypoints across images using projective imagery.

3.5 Sparse Reconstruction: Structure-from-Motion (SfM)

The fourth step, which also by itself is called Structure-from-Motion, is to reconstruct the scene that was taken using 2D images into an initial sparse 3D structure. Using the verified matched keypoints, SfM aims to simultaneously reconstruct the: (a) 3-dimensional scene structure, (b) camera pose and orientation (extrinsic parameters), and (c) intrinsic camera calibration parameters. The intrinsic camera parameters are defined by a camera calibration matrix that includes image scale, skew, and the principal point that is defined as the location on the image plane which intersects the optical axis. Further intrinsic parameters are also required to resolve additional internal aberrations such as distortion on non-pre-calibrated cameras. These intrinsic parameters are either included in the camera's image file format (e.g. EXIF) or will be resolved in additional intermediate steps. After this, a process known as bundle adjustment will be used to produce sparse point-clouds. An example of a sparse reconstruction process is used by COLMAP, a general-purpose SfM-MVS pipeline that makes use of additional steps such as geometric verification and a variant of SfM known as incremental reconstruction (Figure 8). (Schoenberger and Frahm 2016).

3.6 Dense Reconstruction: Multi-View Stereo (MVS)

An additional, post-processing method known as Multi-View Stereo can be applied to the sparse 3d model from SfM in order to generate an enhanced "dense" 3d model. The final output of MVS is a complete 3D scene reconstruction from a collection of images of known intrinsic and extrinsic parameters, which is already resolved through SfM. A variety of MVS algorithms are available but recent variants called clustering views for MVS (CMVS) and patch-based MVS (PMVS) has been observed to perform well against other algorithms (Carrivick, Smith and Quincey 2016). CMVS decomposes the camera poses from bundle adjustment into manageable clusters and PMVS is used to independently reconstruct the 3-dimensional model from these clusters. (Westoby, et al. 2012)

4 SUPERVOXEL SEGMENTATION

After the dense 3d point cloud has been successfully created, the next step is to identify and differentiate with sufficient accuracy the individual fragments of rock on the outermost layers of the muckpile, as well as estimate their volume. For this purpose, a technique known as supervoxel clustering has been considered by the study.

4.1 Point Cloud Library (PCL)

The PCL initiative is a relatively new approach to the 3-dimensional perception of the world through point clouds, having been developed in an age where 3d sensing has become progressively inexpensive. Even though the concept is not new per se, point clouds were generally created using systems that cost thousands of dollars in the past. As the technique has become more accessible in recent years, it has gained popularity in other fields. As such, PCL has been created to present an advanced and extensive approach to 3D perception. Using state-of-the-art algorithms, the library features comprehensive capabilities for filtering, feature estimation, reconstruction, model fitting, and for its main purpose in this study, supervoxel segmentation. (Rusu and Cousins 2011)

4.2 Voxel Cloud Connectivity Segmentation (VCCS)

For the purpose of determining the fragmentation size distribution of a muck pile, the individual fragments located on the outer layers of the muck pile in the generated 3d model to be recognizable from other fragments and measurable in size. To achieve this, a set of algorithms that would be able to split the 3d model into clusters that denote individual fragments is needed. The clusters need to be denoted using several parameters such as color, vector orientation, and Euclidean distance. Voxel Cloud Connectivity Segmentation (VCCS) is one such algorithm that generates supervoxels, which are over-segmented regions in a 3d point cloud. Supervoxels have two key properties: they are distributed evenly across 3d space, and they cannot cross boundaries unless the underlying voxels are spatially connected. VCCS generates supervoxels by incrementally expanding them from a set of seed points distributed evenly in 3d space (Figure 4). The expansion is limited by parameters set by the user, and isolated seeds are filtered out by establishing search radii and removing seeds that do not have an enough number of neighbor voxels. The process proceeds iteratively until the edges have been reached, or the algorithm runs out of neighbors to check (Figure 5). (Papon, et al. 2013)

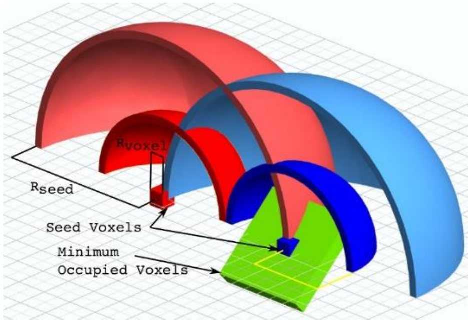

Figure 4. Visualization of supervoxel clustering with user-determined parameters R_{seed} and R_{voxel} (Papon, et al. 2013)

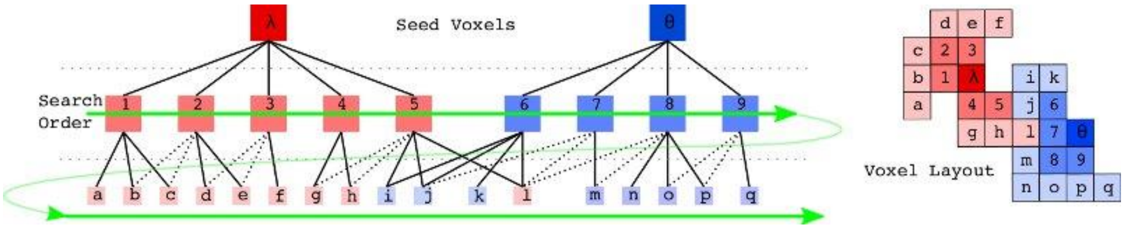

Figure 5. Search order for supervoxel expansion (Papon, et al. 2013)

Applied to a 3d point cloud of a muckpile created from SfM-MVS, VCCS creates clusters (supervoxels) of points which are considered by the algorithm as individual fragments, with each supervoxel then contained within separate bounding boxes. These bounding boxes will then provide volumetric data which will be the basis for acquiring fragmentation size distribution information.

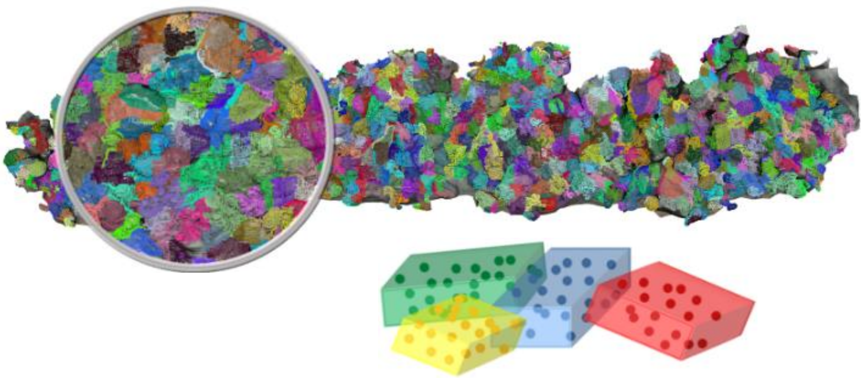

Figure 6. Supervoxel Segmentation applied on a muckpile 3d model and an illustration of bounding boxes

5 METHODOLOGY AND RESULTS

5.1 Scaled Set-up

For this study, the system is tested on a pile of wooden cubes as a scaled-down substitute for a muckpile, for the purposes of a controlled proof-of-concept basic experiment. There are a lot of factors to be considered all throughout the process so using blocks hopes to reduce some of the uncertainty while developing the system before moving on to actual testing. 23 Photos of the wooden blocks are taken using a mobile phone (Galaxy S7 Edge Rear Camera, Auto Mode, 4032x2268 Resolution)

Figure 7. Samples from the image dataset used in the analysis

5.2 Reconstruction using COLMAP

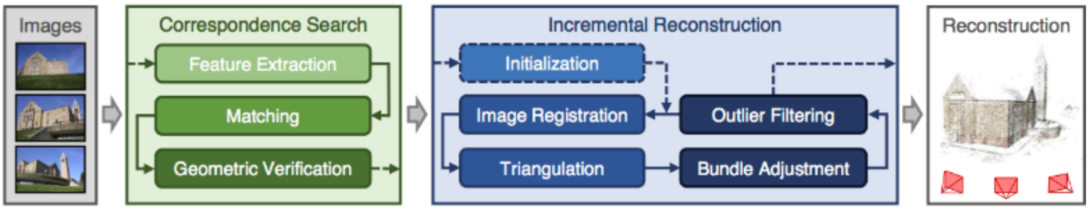

Figure 8. COLMAP's incremental SfM pipeline (Schoenberger and Frahm 2016)

Next, the images are uploaded to COLMAP to reconstruct the wooden blocks in 3D space into a point cloud. Aside from SfM, it also performs the dense reconstruction using PMVS/CMVS, as well as an additional meshing and texturing step. Some cleaning up of stray points were done using MeshLab, an open-source point cloud manipulation tool. (Cignoni, et al. 2008)

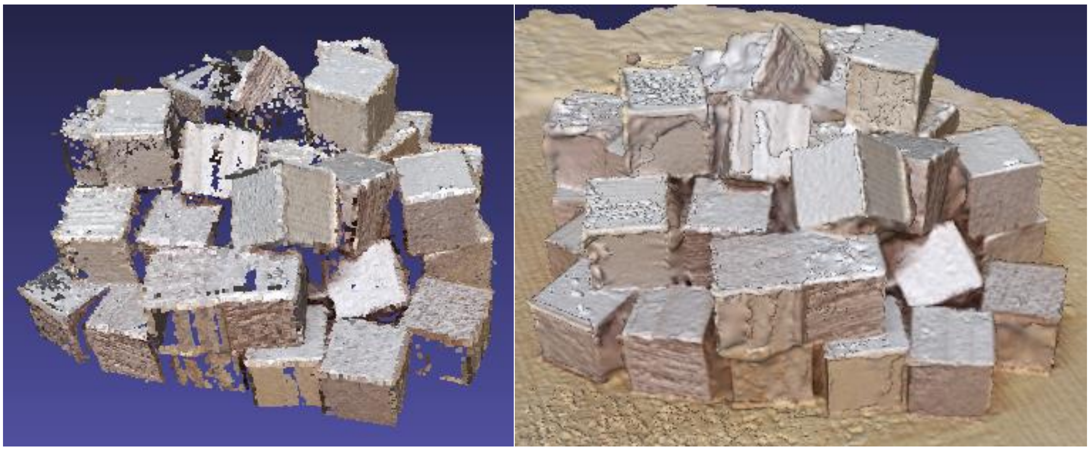

Figure 9. Dense Point Cloud and Meshed 3D Model

5.3 Supervoxel Segmentation

Next, the dense point cloud is uploaded into an Ubuntu environment running inside Windows that has PCL libraries and an executable segmentation program written in C++ (Papon, et al. 2013). The executable will then take the dense point cloud input and run clustering algorithms to create supervoxels and segmentations. For this test, the following parameters were used to run the executable: Seed Resolution = 0.5, Voxel Resolution = 0.01

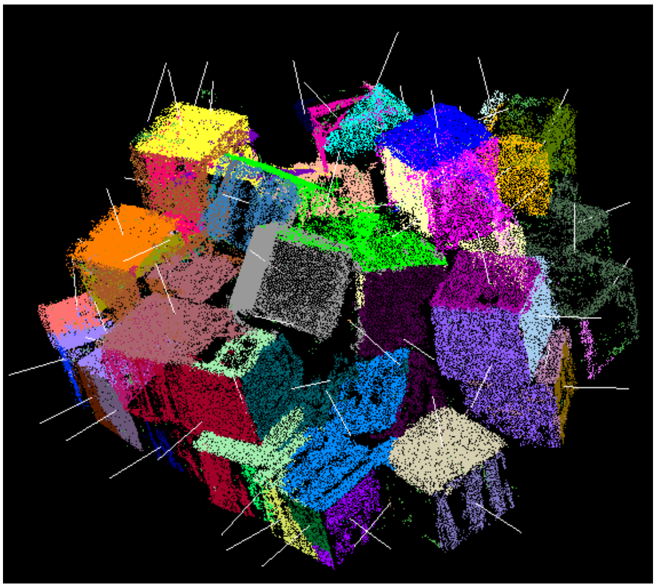

Figure 10. Point cloud output of segmentation

Each individual color corresponds to a single supervoxel, and observing the model tells us that each supervoxel corresponds to one side of a cube. The white lines correspond to the normal, indicating a planar feature. It is however important to note the imperfections present within the segmentation- some supervoxels include multiple sides of one or more cubes, as seen when the color is persistent outside of a single cube side.

5.4 Discussion of Results

The results show that the algorithm is able to recognize individual sides of the wooden cubes and in considering the next steps to the process, it can be tackled two ways-firstly, the individual supervoxels belonging to one wooden cube can be merged using another algorithm that would take into consideration adjacency and angle (taking advantage of the right angles present in cubes) and a bounding box can be formed around the merged supervoxels for volume measurement. Secondly, the parameters can be tweaked further so that each supervoxel would correspond to a wooden cube, and as with the previous method, be wrapped by a bounding box for measurement. It is also of note that upon digitally measuring the dimensions of the supervoxels, they are consistent in measurements across most of the point cloud. A total of 20 sides were measured arbitrarily, which resulted in a fairly even distribution with a standard deviation of 0.984089.

6 CONCLUSION

In this experiment, a pile of wooden cubes was analyzed and translated into a segmented 3d point cloud by taking several images of the subject and using it as an input for Structure-from-Motion and Multi View Stereo, and Supervoxel Segmentation. It can be concluded that the system is able to recognize individual sides of the wooden cubes without the aid of additional data other than the images and the intrinsic properties of the camera. With what has been shown, the author believes that there is merit into continuing this study into using SfM-MVS and Supervoxel Segmentation in the measurement of fragmentation size distribution, and with enough time, a field-ready system can be developed.

REFERENCES

Afum, Bright, and Victor Temeng. 2014. "Reducing Drill and Blast Cost through Blast Optimisation – A Case Study." *UMaT Biennial International Mining and Mineral Conference.* Tarkwa: University of Mines and Technology Ghana. 137-145.

BME South Africa. 2016. "Better blasting can save on power costs." *BME South Africa News.* May 4. Accessed December 19, 2018. http://www.bme.co.za/resources/news/2016/60-better-blasting-can-save-on-power-costs.

Carrivick, Jonathan L, Mark W Smith, and Duncan J Quincey. 2016. *Structure from Motion in the Geosciences.* Wiley-Blackwell.

Cignoni, P., M. Callieri, M. Corsini, F. Dellepiane, F. Ganovelli, and G. Ranzuglia. 2008. "MeshLab: an Open-Source Mesh Processing Tool ." *Sixth Eurographics Italian Chapter Conference.* 129-136.

Fischler, Martin A, and Robert C Bolles. 1981. "Random Sample Consesus: A Paradigm for MOdel Fitting with Applications to Image Analysis and Automated Cartography." *Communications of the American Computing Association* 24 (6): 381-395.

Gräter, Johannes, Tobias Schwarze, and Martin Lauer. 2015. "Robust scale estimation for monocular visual odometry using structure from motion and vanishing points." *IEEE Intelligent Vehicles Symposium (IV).* Seoul.

Häming, Klaus, and Gabriele Peters. 2010. "The structure-from-motion reconstruction pipeline – a survey with focus on short image sequences." *Kybernetika* 46 (5): 926-937.

Lowe, David G. 2004. *Distinctive Image Features from Scale-Invariant Keypoints.* Vancouver: University of British Columbia.

MathWorks. 2018. "Computer Vision System Toolbox: Structure from Motion." Accessed August 6, 2018. https://www.mathworks.com/help/vision/ug/structure-from-motion.html#bu8cmo6-1.

Mikolajczyk, Krystian, and Cordelia Schmid. 2005. "A Performance Evaluation of Local Descriptors." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27 (10): 1615-1630.

Nefis, Mouloud, and Korichi Talhi. 2016. "A Model Study to Measure Fragmentation by Blasting." *Mining Science* 91-104.

Papon, Jeremie, Alexey Abramov, Markus Schoeler, and Florentin Wörgötter. 2013. "Clustering of Pointclouds into Supervoxels - Theoretical primer." Portland: IEEE Conference on Computer Vision and Pattern Recognition.

Robertson, D. P., and R. Cipolla. 2009. "Structure from Motion." Chap. 13 in *Practical Image Processing and Computer Vision*, by M. Varga. John Wiley.

Rusu, R.B., and S. Cousins. 2011. "3D is here: Point Cloud Library (PCL)." *Proceedings - IEEE Conference on Robotics and Automation.* Shanghai.

Schoenberger, Johannes Lutz, and Jan-Michael Frahm. 2016. "Structure-from-Motion Revisited." *Conference on Computer Vision and Pattern Recognition (CVPR).* Zurich.

Snavely, Keith N. 2008. "Scene Reconstruction and Visualization from Internet Photo Collections." Seattle: Univeresity of Washington.

Westoby, M. J., J. Brasington, N. F. Glasser, M. J. Hambrey, and J. M. Reynolds. 2012. "'Structure-from-Motion' photogrammetry: A low-cost, effective tool for geoscience applications." *Geomorphology* (Elsevier) 179: 300-314.