

Volumetric Estimation of Contained Soil using 3D Sensors

Hamza Anwar¹, Syed Muhammad Abbas¹, Abubakr Muhammad¹, Karsten Berns²

¹Dept. of Electrical Engineering, SBA School of Science and Engineering, LUMS, Pakistan
²Dept. of Computer Science, University of Kaiserslautern, Germany

Abstract. Volumetric estimation of contained soil has many potential applications in a variety of tasks such as construction, mining, excavation and landscaping. We propose a novel method of estimating soil volume using 3D sensors that does not require a container model in advance. Our novelty is the realization that estimating soil volume is a measure of accuracy in aforementioned tasks. We demonstrate our assistive system as series of simulated and real-world experiments, along with a study of variation in system parameters to witness its robustness.

1 Introduction

Today's growing demand of automation in large-scale construction and agricultural industry has seen late developments in machine perception. This has even led to high-end sensors and precise actuators in machines, but the need of quantitative analysis of materials operated upon, has not been exclusively addressed. Our interest is pertinent to related endeavors of excavation, landscaping, mining, and canal cleaning in which knowing the desired and the actual amounts of materials (soil, water, and mud) may improve net performance.

We contend that in robotic excavators that use 3D environment sensors, on-the-run volumetric estimation can serve as a good measure of utilization and productivity for the performed excavation. In rough outdoor environments, such as construction sites, achieving autonomy in earthmoving and loading tasks needs highly robust and fail-safe equipment. The problem at hand is to get a measure of how accurate such executions are done in terms of quantitative analysis of soil excavated or loaded. To this end, we have developed an indigenous technique of estimating contained soil using 3D depth sensors. It is becoming prevalent that every real world robot has 3D sensing capability for environment interaction at its core. For that, we use high definition cameras as one-of-many customized stereo systems to generate three dimensional point cloud representation of the excavating container. Because of large variations of exposure and lighting conditions in such outdoor environments, the stereo correspondence algorithm we used was semi-global block matching. After scanning, owing to the existing stereo reconstruction precision, we apply a simple statistical outlier removal filter on the point cloud. Moving on, a small module to chalk out the region

This work has been funded by German Academic Exchange Service (DAAD) for the project titled ALVeDA (Autonomous Land Vehicle for Demining and Agriculture).

of interest in the 3D cloud is implemented. An empty model cloud of container (readily available after first scan) is transformed to align with scanned cloud of the container on-the-fly. This is achieved using iterative closest point (ICP) matching technique. Once aligned, numerical method of rasterizing the space into cells is used and a plane is fit for each cell. Then, individually, volume for each block is numerically calculated by finding the median Euclidean distance between each corresponding pair of planes. The process is repeated for each raster and volume is accumulated to get the total grainy material volume.

Some key points of our approach that add to its relevance with the task of volume estimation need highlights. First, external *model* of the container (CAD drawing, 3D triangular mesh, etc.) is not required because our scheme incorporates generating a model itself, through an initial measurement prior to operation. Second, it is real-time and easily deployable for all 3D perception systems that may see volume estimation as useful information. Third, the solution is simple, low-cost and elegant – not requiring any high-end hardware. And lastly its robustness; as it provides immunity to irregularity in surfaces of contained material (not just soil alone) and the container's form (shape and size) – as long as material surface is completely visible in scans.

2 Related Work

Research work on generic quantitative estimation is found to be sporadic, but in relation to this, there is considerable research on autonomous loading and excavation. Cannon [1], in his thesis on earthmoving with an autonomous excavator, talks about perception enhancements in dig execution. The focus of his work is on planning optimal dig of the excavator to achieve minimum cost – where cost function is determined by volume scooped, energy used and time spent for the single dig. His methodology is to store the shape of terrain prior to digging and, based on bucket's trajectory of scooping, continuously integrate the swept volume. Similar to our approach, the digging and perception are decoupled problems for him. But, his work does not incorporate the alignment issues, nor is it extendable to all sorts of container setups.

Dunbabin [2] also addresses the similar problem of determining *dipper fullness* in rope shovel excavation. He has used two approaches for it: one to use laser-based scanning, and the other to use motor power signals at the *dipper* – measuring quantity of excavated soil. His laser-based sensing determines the maximum height of soil in bucket to calculate volume and ignores the precise soil profile of the container compromising accuracy and portability.

A study on laser scanning for excavation measurement [3] has talked about the registration (alignment) problem in excavations which later leads to discussion of the volume estimation problem. It has produced a solution to this estimation problem by using the depth difference in terrain after excavation but their problem of getting the ground truth i.e. the actual excavated soil volume remained unsolved. Other works by Stentz [4] and Hemami [5, 6] have devised methods to plan dig trajectories that maximize excavated soil volume. However, consideration of the accuracy in achieving the volume requirement in this *planned* scooping has not been properly dealt with, and they have not focused on precision of volume measurements.

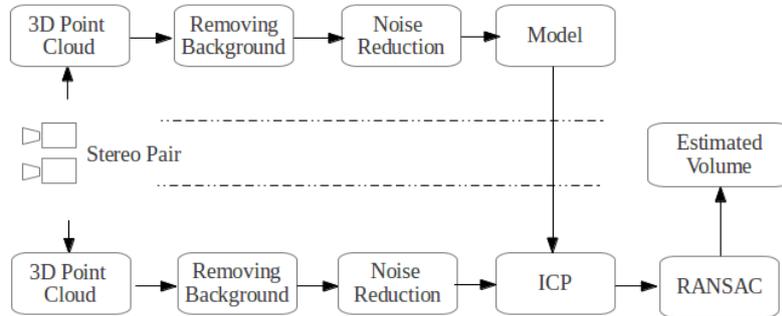


Fig. 1. System level diagram.

3 Methodology

This section describes the whole process of estimation of contained soil step-by-step (Figure 1). The scheme is structured with distinct sensing, perception and estimation parts such that the overall process is configurable to work as a sub-system for a variety of larger systems.

3.1 Sensing

We used stereo camera system as our 3D sensor. Our primary concern in sensing is to have a 3D point cloud. To maximize the information gain and robustness to environmental deviations the precise choice of mechanical system design and vision algorithms in use deserves extreme importance.

Assembly: Our application is designed for estimation of quantity of soil in containers that are in the range of 2-3 meters in distance from the 3D sensors. In this context, we employed a self-built low-cost stereo system (Figure 2). We used USB powered high-definition pair of webcams and constructed a firm housing for the pair. Both cameras face front-parallel with a fixed baseline of 10 cm.

Calibration: We use the calibrated-camera stereo calibration method [7] using a set of chessboard images, to get both the intrinsic and extrinsic parameters of the cameras.



Fig. 2. Stereo camera pair



Fig. 3. Real bucket 2/3rd filled



Fig. 4. Disparity image 2/3rd filled

This includes the camera *reprojection* matrix [8], later used in 3D reconstruction.

Matching: We use Semi-Global Block Matching (SGBM) [9] as our correspondence algorithm with parameters minimum disparity 0, number of disparities 80 and block window size of 9x9 pixels. This algorithm was chosen on account of its scalability, real-time processing, and the robustness of the cost function for block matching it employs. Using this technique and the pre-computed

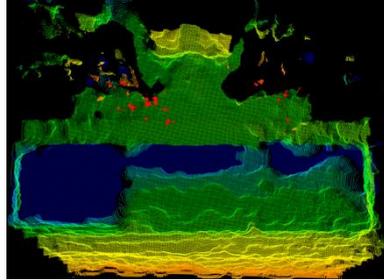


Fig. 5. 3D point cloud 2/3rd filled.

reprojection matrix we generate the disparity map of the scene. Stereo matching process runs with one snapshot of *current* scene (Figure 3) and creates a disparity image having the required depth information of the whole container (Figure 4).

Point Cloud: With the disparity image, and camera intrinsic matrix, we *reproject* all points to 3D world to get a Point Cloud (Figure 5); this is done by using Point Cloud Library [10] and OpenCV [8].

Background & Outlier Removal: After we obtain the environment's point cloud (Figure 5), we subtract the unnecessary background (the region of space more than maximum expected distance of the container) and then apply a statistical filter to remove outliers from the 3D cloud. The co-ordinate system origin here is the left camera's center with z-axis along the camera axis (depth).

3.2 Container Modelling

After getting the environment's 3D scan, we come to the perception stage of the algorithm. The foremost thing needed for volume estimation in our approach is the model of the container we are estimating the soil in. There are plenty of ways to accomplish this, including a CAD drawing available from the manufacturer of the container, or a database of standard containers in varying shapes and sizes. But to make our process independent of such initial data we make our own model of the container.

The container model is generated only once. We employ the aforementioned 3D sensing technique on a stereo image pair in which our container is empty and completely visible. This will serve as a reference cloud when comparing it with the *current* cloud for estimating change in volume contained.

Once we have this model, we manually mark the four corners (Figure 6) of the

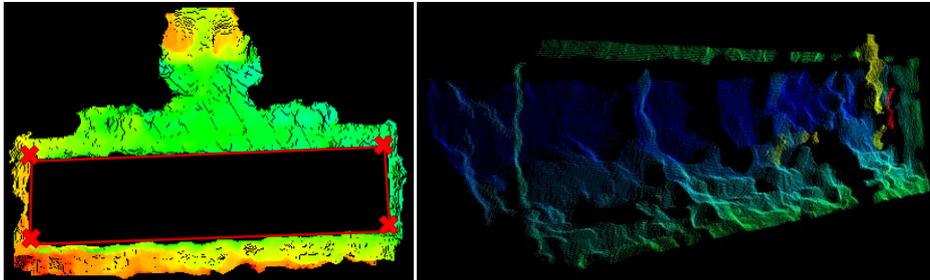


Fig. 6. Left: Outer part of model for alignment and red marks to manually choose the container boundary. **Right:** Interest area separated from rest of the model for volume estimation.

container in the image. This way we divide our model into two point clouds: the *outside* of the container (used in alignment) and other, comprising the *inside* of the model for volume measurement. Manual marking is acceptable as the model is only created once for a particular container.

3.3 Alignment

Once we have the empty container model in the form of two complimentary point clouds, we are ready to take the new scan of the filled container. The newly acquired point cloud will most likely not have the container at the exact same position as the model, due to bucket motion during excavation. Thus, we need to align them.

One requirement (that is practically possible) is that the container is not expected to have arbitrarily changed its position while filling up because for a structured arrangement of the sensors mounted over the vehicle, the container is brought, more or less, to the same position when its volume needs to be estimated. Note that this requirement does not apply to pose changes since the algorithm explained below can handle arbitrary changes in orientation.

The alignment is achieved by Iterative Closest Point (ICP) matching [11, 12] being *current* scan as the target and the *outside* model as the source with which target points are to be aligned. Maximum iterations of 200 are used for ICP and 0.05 meters is the maximum correspondence distance. RANSAC outlier rejection threshold was set to 1 and transformation epsilon to 10^{-8} . It gives us a 3D transformation matrix existing between the two clouds by which we transform the current cloud. After applying the transformation on every point in a *current* point cloud, the resulting points are exactly aligned to the model (Figure 7). This enables us to filter the region of interest from the *current* cloud which now overlays directly on the point clouds of the model.

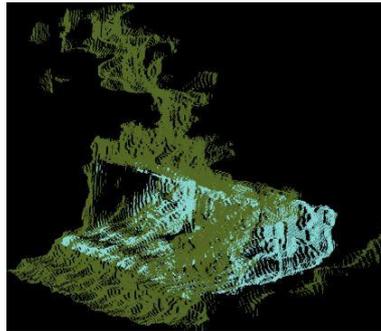


Fig. 7. *Blue*: Model point cloud.
Green: Aligned *current* cloud.

3.4 Volume Estimation

Finding the volume is the final step of this process. In this, we compartmentalize both the model and the *current* clouds into two 3D grids respectively. The grid size is manually chosen based on a variety of parameters – density of the point cloud, the size and capacity of the container box, the granularity of the material (soil), and of course, the requirement precision and accuracy of measurement.

Since our *current* and model clouds are both aligned we can directly compare the points in their grid elements. Inside every raster, we fit a plane in each of the two corresponding point cloud boxes. The plane coefficients are approximated by running RANSAC (RANdom SAMple Consensus) over the data [13]. With this we get the two plane equations.

For missing cells i.e. holes with not enough points for plane fitting, we fill the region of space by assigning it the average plane coefficients of the nearest neighboring

rasters. Next, we find the two median points of the two planes. While finding the height of soil in a grid cell, we use the median point of the plane instead of using the mean point to reduce outlier weightage. Euclidean distance between these two points is computed – this is the height of soil for a particular cell. The height along with the knowledge of the grid size in units of length are used to calculate the volume enclosed by the raster. This process of numerical integration is repeated over the entire grid to get the total aggregated volume of contained soil.

4 Experiments in a Simulated Environment

We have used *Finroc* based *SimVis3D* simulation tool [14, 15] to simulate the bucket excavator’s control environment. This environment has relevant characteristics that facilitate our work (Figure 8) such as soil modeling using Newton Dynamics. It also has the capability to simulate various kinds of sensors.

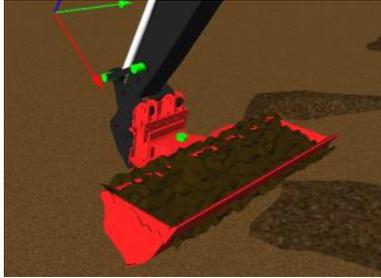


Fig. 8. Simulation snapshot of bucket excavator in SimVis3D.

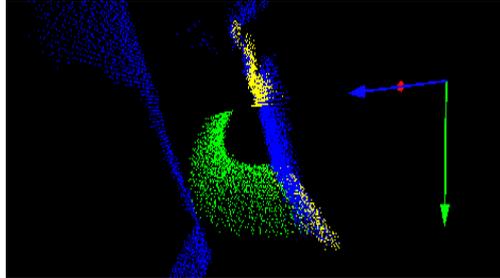


Fig. 9. *Green* and *Yellow*: Model point clouds aligned with *current* point cloud (*Blue*)

In the simulated experiments [16], the time-of-flight camera has sensor noise modelled by a zero mean Gaussian distribution with a precision of 4cm; and angular resolution of 0.5°. The bucket bounding box dimensions are setup as 2.05m x 0.54m x 0.6m. We run a series of experiments by varying grid sizes and soil quantity. In simulation, the bucket is filled up by varying soil height from its base. The surface profile is given a realistic look by perturbing the height at each grid point randomly (Figure 8). In Figure 9, *current* cloud is transformed to align with the model [17]. It is ready for grid overlay and volume estimation. Table 1 shows the experiments results.

Table 1. Filled bucket with soil quantities 0 to 520 x 10³ cm³ – grid cell size of 10cm

Soil surface height from base (cm)	Actual Volume (10 ³ cm ³)	Estimated Volume (10 ³ cm ³)	Error in Estimation (%)
0	0.0	17.6	∞
10	28.8	26.2	9.03
20	99.1	90.8	8.32
30	188.9	183.3	2.97
40	294.6	285.3	3.15
45	348.9	336.7	3.49
50	403.1	384.4	4.65
55	459.5	438.3	4.60
60	520.8	498.1	4.34

In simulation results, there is an estimated volume of 17606 cm³ against 0 cm³ of actual volume. This is due to combination of sensor noise and misalignment errors. Note that this error is small as the container is of much larger size: having capacity of 2.05 x 0.54 x 0.6 = 0.6642 m³ = 664200 cm³. From simulated experiments, we have achieved an accuracy of over 91% in estimating the correct volume of contained soil in worst scenario case. Results of experiments with varying grid size for a fixed quantity of soil are shown in Table 2.

Table 2. Varying grid size for fixed amount of soil i.e. fully filled (height of 0.6m)

Grid cell square size	Empty cells / Total cells	Estimated Volume (10 ³ cm ³)	Actual Volume (10 ³ cm ³)	Error in estimation (%)
15cm	0/56	520.8	454.4	12.73%
12cm	0/85	473.2	454.4	3.96%
10cm	3/126	498.1	454.4	8.75%
9cm	3/138	498.2	454.4	8.77%
7cm	40/232	501.5	454.4	9.38%
5cm	77/451	511.3	454.4	11.10%
3cm	1070/1156	447.1	454.4	1.67%

Also, it is evident that we can achieve higher precision through variation in grid cell size. Smaller grid sizes increase missing data per cell but by filling cells with an average, we get lesser mean percentage error. Note that the increase in number of empty cells does not mean that the missed data is increased, instead the total number of data points remains same, and it is just the number of points present in a particular cell. The actual volume of soil is calculated by fitting a plane at 0.6m height from the bucket base with zero sensor noise and no misalignment error and calculating the volume by numerical integration with least raster size from the known bucket model.

5 Real-world Experiments

After getting encouraging results in simulation, we conducted experiments on a real bucket excavator. One of the many physical arrangements we tested for our setup is shown in Figure 10.

We filled the bucket with a measured quantity of soil using a smaller bucket of known capacity – this was our source of the volume ground truth for comparison. We conducted experiments at different stages during our filling of 305 liters of mud. Our sensing system showed great results for generating dense point clouds despite it being our custom made stereo pair. A complete set of data collected for all three stages is shown in Figure 11. The effect of varying quantity of soil is shown in Table 3.

Table 3. Results of bucket filled up with soil of up to 305400 cm³ – grid cell size of 1cm

Quantity of Soil in Bucket	Actual Volume (10 ³ cm ³)	Estimated Volume (10 ³ cm ³)	Error in Estimated Volume (%)
Full	305.4	307.2	0.57
Two-thirds	215.8	220.7	2.23
One-thirds	146.6	169.1	15.4
Empty	0.0	27.1	∞



Fig. 10. Physical arrangement of real world setup.

Table 4. Grid sizes with constant soil quantity (full bucket; total data points: 76480).

Grid cell square size	Empty cell / Total cells	Avg. points per grid cell	Estimated Volume (10^3 cm^3)	Actual Volume (10^3 cm^3)	Error in estimation (%)
12cm	7/90	413.4	387.8	305.4	26.9
9cm	27/168	244.8	391.9	305.4	28.3
7cm	26/248	153.5	365.8	305.4	19.8
5cm	38/473	79.8	358.0	305.4	17.2
4cm	48/742	50.3	353.6	305.4	15.8
3cm	145/1349	29.2	365.3	305.4	19.6
2½cm	153/1870	20.1	352.7	305.4	15.5
2cm	327/2968	13.0	357.5	305.4	17.1
1½cm	762/5217	7.4	352.5	305.4	15.4
1cm	8673/11605	4.4	307.2	305.4	0.57

Table 4 shows the effect of variation in grid size for fully filled bucket while measuring its volume. These tests were performed in an outdoor setting that is similar to realistic commercial vehicle operations. Thus the achieved accuracy demonstrates the viability of our proposed solution in real construction and industrial operations.

6 Discussion

It is evident from above tables that there is a trade-off between the accuracy of estimation and fraction of empty cells. For very small grid cells we do not have enough points for fitting planes, hence estimation starts to over-fit. In the real-world experiments, a grid cell size of ~1cm causes the error to drop significantly, but note that the number of cells without enough points to fit a plane increases to 74% (8673/11605). The remaining cells have 4.4 points each on average. Thus, we didn't go beyond 1cm for estimation. Remember that the empty cells do not imply any missing data because the overall point cloud is the same.

In this paper, we concentrated on the implementation and validation of a particular approach. For the alignment task, ICP assumes no significant variation in the container position compared to its model's position. An extension of this work can be to use forward kinematics for relative positioning of the sensor and the container. Knowledge of the transformation of co-ordinate systems existing at the joints of machine may help in a much more accurate alignment.

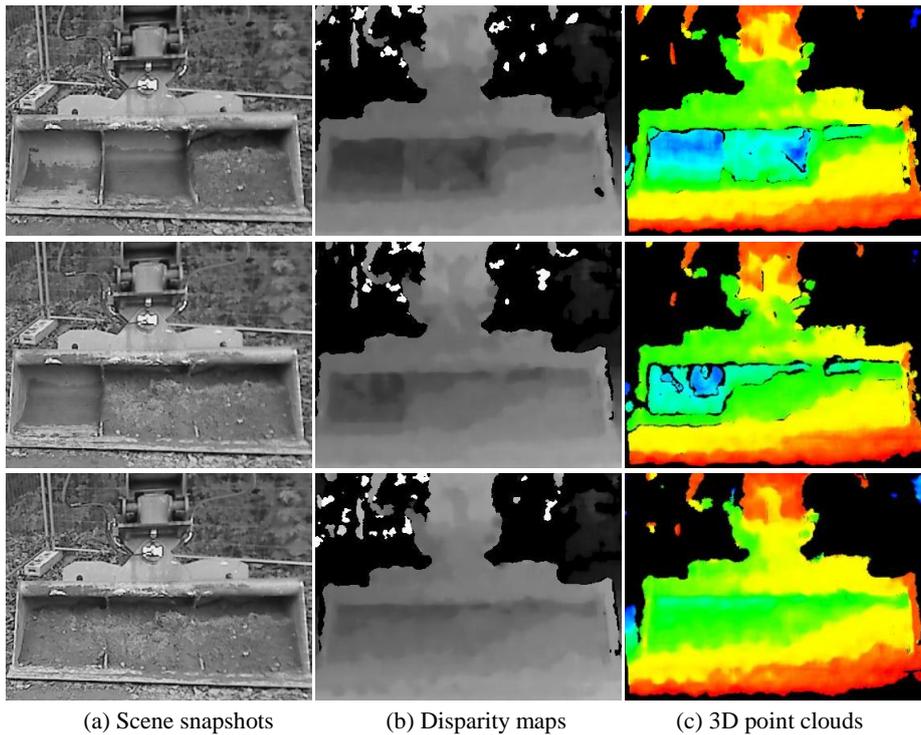


Fig. 11 Three stages of filling mud and estimating volume.

7 Conclusion

In this paper we have presented a novel method of estimating the volume of contained soil using stereo vision. Major advantage of this approach is the use of 3D point cloud as a metric for the task and solution to the registration problem in measurements. Our approach is also applicable for various contained materials, and for any form (shape and size) of the container. Having the capability to generate its own container model makes it easier to deploy. With our approach in simulation and real-world bucket excavator experiments, we have illustrated high accuracy of estimation. This also verifies our technique's robustness to variations in environment conditions.

Acknowledgements

We are grateful to Daniel Schmidt (TU-KL) for help in conducting simulations and on-field excavations; and Ali Musa Iftikhar (LUMS) for initial findings.

References

1. Cannon, H.: Extended Earthmoving with an Autonomous Excavator. Masters Thesis, Carnegie Mellon Robotics Institute, CMU-RI-TR-99-10, 1999.
2. Dunbabin, M., Corke, P.: Autonomous Excavation Using a Rope Shovel. *Journal of Field Robotics* 23(6-7):379-394, 2006.
3. Cheok, G., Lipman, R., Witzgall, C., Bernal, J., Stone, W.: Field Demonstration of Laser Scanning for Excavation Measurement. In *Proceedings of ISARC*, 2000.
4. Stentz, A., Bares, J., Singh, S., Rowe, P.: A Robotic Excavator for Autonomous Truck Loading. *Autonomous Robots* 7(2):175-186, 1999.
5. Hemami, H.: Study of Bucket Trajectory in Automatic Scooping with LHD Loaders. *Transactions Institution Mining, Metallurgy A*: 102, 1992.
6. Hemami, A., Hassani, F.: An Overview of Autonomous Loading of Bulk Material. In *Proceedings of ISARC*, 2009.
7. Hamzah, R.A., Abd Ghani, S., Kadmin, A., Aziz, K.: A practical method for camera calibration in stereo vision mobile robot navigation. *IEEE Student Conference on Research and Development (SCORED)*: 104-108, 2012.
8. Bradski, G., Kaehler A.: *Learning OpenCV: Computer vision with the OpenCV library*. O'Reilly Media, 2008.
9. Hirschmüller, H.: Stereo processing by semi-global matching and mutual information. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30(2): 328-341, 2008.
10. Rusu, R.B., Cousins, S.: 3D is here: Point Cloud Library (PCL). *IEEE International Conference on Robotics and Automation (ICRA)*. Shanghai, 2011.
11. Besl, P., McKay, N.: A Method for Registration of 3-D Shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 14(2): 239-256, 1992.
12. Zhang, Z.: Iterative Point Matching for Registration of free-form curves and surfaces. *International Journal of Computer Vision* 13(2): 119-152, 1994.
13. Martin, A.F., Robert, C.B.: Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Comm. of the ACM* 24(6): 381-395, 1981.
14. Reichardt, M., Föhst, T., Berns, K.: On Software Quality-motivated Design of a Real-time Framework for Complex Robot Control Systems. *7th International Workshop on Software Quality and Maintainability (SQM)*. Genova, 2013.
15. Wettach, J., Schmidt, D., Berns, K.: Simulating Vehicle Kinematics with SimVis3D and Newton. *2nd International Conference on Simulation, Modeling and Programming for Autonomous Robots*. Darmstadt, Germany, 2010.
16. Schmidt, D., Berns, K.: An autonomous excavator for landscaping tasks. *Proceedings for the joint conference of ISR and ROBOTIK 2010*: 819-826. VDI Dusseldorf, 2010.
17. Zolynski, G., Schank, C., Berns, K.: Point Cloud Gathering for an Autonomous Bucket Excavator in Dynamic Surroundings. *Proceedings of the Commercial Vehicle Technology Symposium*. Verlag, 2012.