

COMBINED VISUAL EXPLORATION OF 2D GROUND RADAR AND 3D POINT CLOUD DATA FOR ROAD ENVIRONMENTS

J. Wolf^{1,*}, S. Discher¹, L. Masopust², S. Schulz², R. Richter¹, J. Döllner¹

Hasso Plattner Institute, Faculty of Digital Engineering, University of Potsdam, Germany

¹ (johannes.wolf, soeren.discher, rico.richter, juergen.doellner)@hpi.de

² (leon.masopust, sebastian.schulz)@student.hpi.de

KEY WORDS: Ground Penetrating Radar, 3D Point Cloud, Infrastructure Maintenance, Anomaly Inspection, Visualization Tool

ABSTRACT:

Ground-penetrating 2D radar scans are captured in road environments for examination of pavement condition and below-ground variations such as lowerings and developing pot-holes. 3D point clouds captured above ground provide a precise digital representation of the road's surface and the surrounding environment. If both data sources are captured for the same area, a combined visualization is a valuable tool for infrastructure maintenance tasks. This paper presents visualization techniques developed for the combined visual exploration of the data captured in road environments. Main challenges are the positioning of the ground radar data within the 3D environment and the reduction of occlusion for individual data sets. By projecting the measured ground radar data onto the precise trajectory of the scan, it can be displayed within the context of the 3D point cloud representation of the road environment. We show that customizable overlay, filtering, and cropping techniques enable insightful data exploration. A 3D renderer combines both data sources. To enable an inspection of areas of interest, ground radar data can be elevated above ground level for better visibility. An interactive lens approach enables to visualize data sources that are currently occluded by others. The visualization techniques prove to be a valuable tool for ground layer anomaly inspection and were evaluated in a real-world data set. The combination of 2D ground radar scans with 3D point cloud data improves data interpretation by giving context information (e. g., about manholes in the street) that can be directly accessed during evaluation.

1. INTRODUCTION

Mobile mapping techniques have been established in various environments (Li, 1997) and become more and more common for infrastructure maintenance purposes (Li et al., 2017). Municipalities and private organizations are capturing geospatial data with multi-scanner systems, enabling detailed analysis of urban environments, large facilities, and infrastructure networks (Airfield Inventory, 2018).

Remote sensing equipment, such as LiDAR scanners, can be mounted on top of arbitrary cars and enables high precision scans of road surfaces, curbstones, and road space assets (Jaakkola et al., 2008). 3D point clouds resulting from such scans are highly detailed representations of the scanned environment and are used as “digital twins” in a variety of application fields (Olsen and Kayen, 2013, Kumar et al., 2016). Depending on additionally captured attribute data, 3D point clouds can be colored with RGB information or intensity values, thus creating data sets intuitively visually explorable even by untrained users within GIS programs and specially designed 3D point cloud visualization tools, like shown in Figure 1.

Ground penetrating radar (GPR) has been used for below-ground analyses for several decades. Ground penetrating radar scanners can measure material properties several meters below ground, creating insights about the non-visible foundation of roads and pathways (Davis and Annan, 1989). A typical visualization of a ground penetrating radar data B-scan is shown in Figure 2. Ground penetrating radar scanners can also easily be mounted onto scanning vehicles, adding an additional data source for the region not accessible by LiDAR scanning (Mobile GPR, 2017).



Figure 1. Visualization of a 3D point cloud of a road environment.

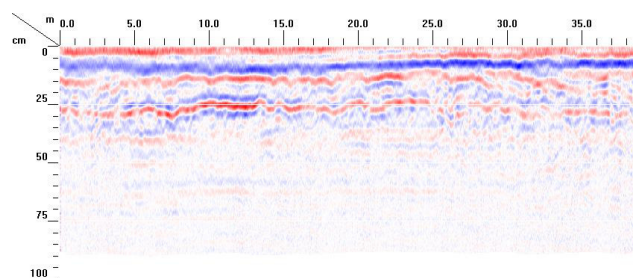


Figure 2. Visualization of a GPR B-scan.

*Corresponding author

The combination of above-ground 3D point clouds and below-ground 2D radar scans enables a more extensive analysis of road environments by using two combined sources instead of evaluating each on their own. A common use case for ground penetrating radar data inspection is to find certain areas with increased chance of developing pot-holes (Huston et al., 2000). By adding road surface information from the 3D point cloud, false positives like manholes can easily be distinguished from other anomalies in the road's subsoil.

2. RELATED WORK

Benedetto et al. give an overview of how ground penetrating radar can be used for road inspections (Benedetto et al., 2017). They discuss in detail which processing techniques can be used to analyze the pavement condition. Evans et al. summarize the abilities of ground penetrating radar for general pavement investigations (Evans et al., 2008). Saarenketo and Scullion explicitly list localization of sinkholes and crack growth monitoring in their report about ground penetrating radar applications on roads and highways (Saarenketo and Scullion, 1994). They further describe possible soil and road structure evaluations and required data interpretation techniques (Saarenketo and Scullion, 2000). Giannopoulos describes how ground penetrating radar data can be visualized (Giannopoulos, 2005). In addition to two-dimensional profiles, he shows an example for a three-dimensional visualization by rendering the three principal planes of a cuboid which is holding the data. Usually, two-dimensional ground penetrating radar profiles are shown individually, but multiple scans can be placed next to each other to create spatial feeling for the data (Geo Radar: 3D and GPR, 2005).

Discher et al. and Eitel et al. describe the relevance of 3D point clouds for a large number of geospatial applications (Discher et al., 2019, Eitel et al., 2016). Puente et al. review available mobile terrestrial laser scanning systems, showing that LiDAR is widely used on various mobile platforms for data acquisition leading to users being enabled to “experience and work directly with real-world conditions by viewing and manipulating rich point clouds” (Puente et al., 2013). Biasion et al. describe mobile laser scanning applications for environment analysis during disaster management (Biasion et al., 2005). Several authors discuss the automated analysis of 3D point clouds: Per-point surface category information can be derived by analyzing a 3D point cloud's topology (Chen et al., 2017) and by applying deep learning concepts (Boulch et al., 2017). In turn, derived surface category information can be used to reconstruct three-dimensional models of specific buildings or infrastructure assets as shown by Teizer et al. (Teizer et al., 2005).

A general overview of rendering techniques for 3D point clouds is presented by Gross and Pfister (Gross and Pfister, 2011). While *photorealistic* approaches (Schütz and Wimmer, 2015a, Preiner et al., 2012) aim to reduce typical artifacts (e.g., visual clutter or a holey surface representation) by rendering points with an appropriate size and orientation, *non-photorealistic* techniques (Simons et al., 2014, Zhang et al., 2014) deal with the inherent fuzziness of a 3D point cloud by emphasizing edges and structures. All those rendering techniques can be seamlessly integrated into our rendering system, as demonstrated with the *eye dome lighting* technique originally introduced by Boucheny (Boucheny, 2009). Focus+context visualization techniques have been widely discussed in the context of mesh-based models (Vaaranemi et al., 2013, Elmqvist and Tsigas, 2008), ranging

from static visibility masks (Sigg et al., 2012) to interactive lenses (Trapp et al., 2008). Discher et al. apply such techniques to 3D point cloud depictions, enabling users to highlight task-relevant but occluded objects and structures (Discher et al., 2017). Similar to our approach they apply multi-pass rendering based on G-buffers; however, they don't combine 3D point clouds and additional geodata. To render 3D point clouds of any size, state-of-the-art out-of-core rendering concepts decouple rendering efforts from data management. Exemplary systems (Martinez-Rubi et al., 2016, Goswami et al., 2013, Rusinkiewicz and Levoy, 2000) subdivide 3D point clouds into small, representative subsets that are suitable for real-time rendering. Recent approaches combine out-of-core and web-based rendering concepts, allowing for an ubiquitous visualization on a diverse range of client devices (Discher et al., 2018, Schütz and Wimmer, 2015b, Butler et al., 2014). While the current implementation of our rendering system focuses on a local visualization on individual desktop computers, we plan to integrate web-based rendering concepts as part of future work.

3. DATA CHARACTERISTICS AND REQUIREMENTS

State-of-the-art in-situ and remote sensing technology allows for generation of digital twins of real-world surfaces at minimal time and cost. Its technological basis are (1) active sensors (e.g., LiDAR) emitting electromagnetic radiation to directly measure the distance between surface and sensor to generate range data (Eitel et al., 2016) as well as (2) passive sensors (e.g., digital cameras) relying on natural radiation, most notably sunlight, to generate a series of images that serves as input for dense image matching algorithms to derive 3D information (Remondino et al., 2013). While passive sensors allow for a faster raw data collection, they also require a computation-intensive post processing, whereas active sensors directly produce 3D point clouds. Regarding data quality, both, active and passive sensors allow for the generation of highly detailed 3D point clouds featuring point densities of up to a few micrometers for individual objects (Discher et al., 2019). At a larger scale, albeit at reduced point densities, 3D point clouds of entire building complexes, cities, or countries can be efficiently generated by placing sensors at key positions (i.e., *terrestrial capturing*) or by attaching sensors to moving vehicles such as cars and trains (i.e., *mobile mapping*) or UAVs and planes (i.e., *airborne capturing*), resulting in massive data sets comprising billions of points and hundreds of gigabytes of data.

Ground penetrating radar (GPR) is used for below-ground inspection by emitting electromagnetic waves into the ground and receiving the reflected signal from pavement and soil. The ground's structure impacts propagation of the emitted signal, thus the returned signal provides information about the materials' conditions. Radar antennas for road inspection are usually mounted on scanning vehicles, which can drive along with uninterrupted traffic. The antennas are hovering above ground and their distance to the surface slightly changes while driving. A time zero correction is usually part of the sensor calibration to account for these changes. After capturing, the radar scan data is usually analyzed in the form of B-scans. They are a consecutive sequence of individual measurements (A-scans), in this case along the driving path of the vehicle. The x-axis in Figure 2 represents the traveled distance, the y-axis shows the results of the individual scans. Displayed in a two-color representation, e.g., with red and blue, the direction of the signal's amplitude is represented by hue and its size by saturation. Mobile mapping vehicles use *Global Positioning System (GPS)* to track their current position. The information

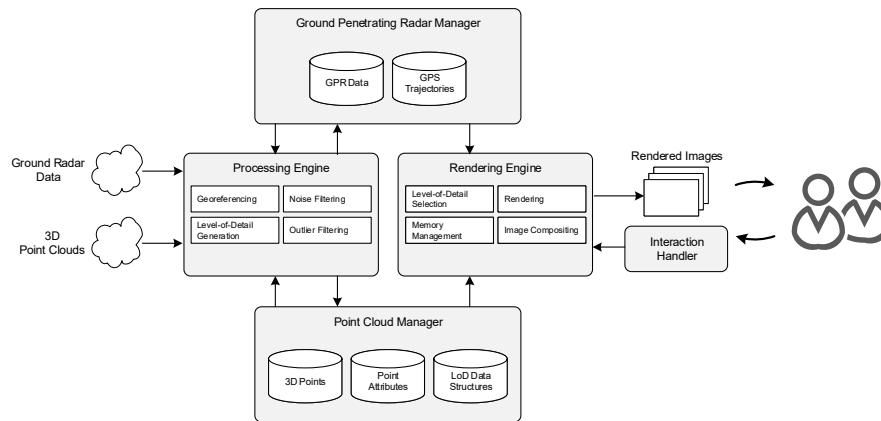


Figure 3. System overview showing data flow integration, processing, visualization, and interaction components.

is stored as pairs of timestamp and location, which together form the trajectory of the measurement path. Using this information and the knowledge about the fixed positions of LiDAR scanners and ground penetrating radar scanners on the scanning vehicle, the different data sets can be collocated in exact spatial relation to each other.

Based on those data characteristics, we have identified the following requirements that need to be addressed for a combined visualization of 3D point clouds and GPR data:

- R1** No limitations regarding used acquisition methods, as well as the number, scale and size of the data sets. The latter is especially important for 3D point clouds, as those can easily contain hundreds of gigabytes of raw data.
- R2** Correct positioning of GPR data and 3D point clouds into a homogeneous spatial reference system.
- R3** Occlusion-free visualization of individual GPR B-scans within a GPR data set.
- R4** Visual filtering and highlighting techniques to enable a focused inspection of areas of interest, that can be defined at runtime.

4. SYSTEM OVERVIEW

We addressed the aforementioned requirements in the design and implementation of our rendering system (Fig. 3). It consists of the following major components:

4.1 Point Cloud Manager

We sort all 3D point clouds into a single, homogeneous spatial data model. Access to that model is handled by the *point cloud manager*, storing spatial information together with additional per-point attributes (e. g., color values). *Level of detail* (LOD) representations such as quadtrees (Gao et al., 2014), octrees (Elseberg et al., 2013), or kd-trees (Richter et al., 2015) are required to efficiently access arbitrary subsets of any size, which is essential for an efficient rendering of massive 3D point clouds (**R1**). These LOD representations are generated by the *processing engine*.

4.2 Ground Penetrating Radar Manager

Fulfilling a similar role as the point cloud manager, the *ground penetrating radar manager* enables efficient access to GPR data, which we store in combination with simultaneously captured GPS trajectories. Based on that information, GPR data can be positioned precisely within 3D point clouds of the corresponding area (**R2**). Individual B-scans within a GPR data set can be accessed separately (**R3**). However, we opted against using LOD representations, since the size of the GPR data sets evaluated in the context of this paper was neglectable by comparison (i. e., 25.7 MB raw data per B-scan).

4.3 Processing Engine

The *processing engine* conducts different pre-processing operations on given data sets, ranging from (a) georeferencing data (e. g., by combining GPR data and GPS trajectories), over (b) data cleaning (e. g., filtering of noise and outliers in 3D point clouds) to (c) generating LOD representations. The processing engine enables users to run and schedule multiple operations in parallel. Results of those operations are automatically stored by the point cloud manager and ground penetrating radar manager, respectively.

4.4 Rendering Engine

The rendering engine is responsible for providing an interactive, combined visualization of 3D point clouds and GPR data. To that end, we apply multi-pass rendering utilizing G-buffers (Saito and Takahashi, 1990) for image-based compositing:

For each data type the corresponding manager is queried, retrieving only data subsets that are relevant for the current view and task. For example, we apply view frustum culling and detail culling to reduce the number of rendered points to an amount manageable by available CPU and GPU capabilities. Regarding GPR data, the individual B-scans are loaded on demand. The retrieved subsets are rendered into separate G-buffers (i. e., specialized frame buffer objects combining multiple 2D textures for, e. g., color, depth, and normal values). A compositing pass merges the separate G-buffers, enabling the application of image-based post processing effects that facilitate visual filtering and highlighting, ranging from emphasizing depth differences in 3D point clouds (e. g., eye dome lighting (Boucheny, 2009)) to more sophisticated focus+context visualization techniques (e. g., interactive lenses).

Changes to the current render configuration (e.g. regarding applied post processing effects) can be made at runtime via the *interaction handler*.

4.5 Interaction Handler

The interaction handler is responsible for updating the visualization according to user interactions. In particular, users can (1) change view position and angle, (2) select B-scans they want to focus on, (3) select and configure applied post-processing effects, and (4) define areas of interest for highlighting (R4).

5. VISUALIZATION TECHNIQUES

The combined visualization of 3D point clouds and GPR data is based on two major user interface components: A *3D scene view* and a *2D user interface*.

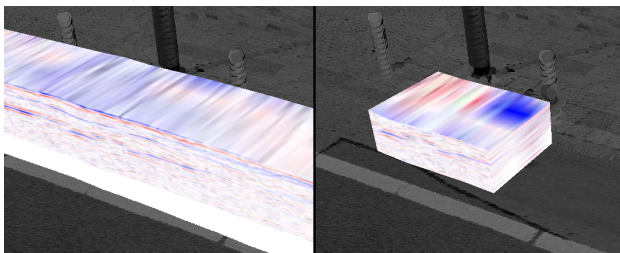


Figure 4. Cuboid rendered onto the GPS trajectory (left). Vertical and horizontal slicing can be used to explore data inside the cuboid (right).

5.1 3D Scene View

The first step towards integrating GPR data and 3D point clouds into a single visualization is projecting each B-scan onto the captured GPS trajectory (R2). To prevent different B-scans from occluding each other, individual B-scans can be hidden dynamically (R3). Each GPR data set is represented by a cuboid-like structure, covering the amount of space scanned by the GPR sensors, that is rendered onto the GPS trajectory (Fig. 4). To fill the area in between the B-scans, their values are interpolated. A 3D-texturing approach guarantees the possibility of slicing the cuboid – both vertically and horizontally – as well as moving it along the trajectory. Thus, the cuboid can be restricted to specific areas of interest, thus, facilitating visual filtering and highlighting (R4).

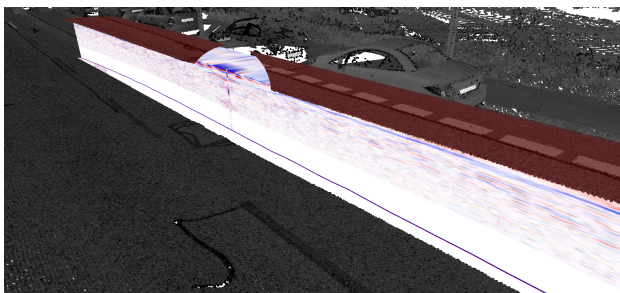


Figure 5. Points under the cuboid are elevated and highlighted. An interactive lens shows the cuboid's surface below the points.

We increase visibility and usability of the cuboids by raising them onto the GPS trajectory instead of leaving them below ground level (R3). To keep the spatial context, the points located originally above the cuboid are translated alongside and highlighted

for a better contrast to non-translated points (Figure 5). Furthermore, to enable a direct view onto the cuboid's surface, we provide an interactive lens that hides points around the cursor. The visualization can be switched to instead only show the points above the cuboid around the cursor, for focusing on the GPR data, while keeping once more the spatial context (R4).

5.2 2D User Interface

To facilitate an in-depth exploration of the GPR data, a supplementary widget is provided visualizing all B-Scans in full length in 2D (Figure 6) and enabling configuration of the 3D scene view.

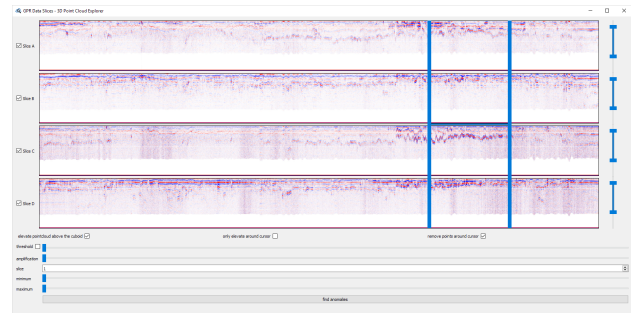


Figure 6. 2D User Interface for GPR data with cropping and thresholding options.

First, users may change how the cuboid of the given GPR data set is rendered. This includes (1) setting its elevation relative to ground level, (2) cropping to specific start and end points, (3) cropping to specific minimum and maximum radar scanning heights, and (4) hiding specific B-scans altogether. Cropping to specific start and end points enables users to move both, the cuboid and the corresponding slice in the 2D view, back and forth. Doing so moves the camera position in the 3D view accordingly, ensuring that the view is always centered on the cuboid (R4).

Second, users may change, how textures are generated from a GPR data set. While the input data includes raw information about the reflected signal picked up by the receiver, the generated textures show the amplitudes of these values, with positive and negative values color coded in red and blue. Users may change how much these values should be amplified, since their range can vary greatly. As an example, a much stronger amplification should be applied when exploring parts of the data with small differentials. Furthermore, parts featuring drastic changes (i.e., most often points of interest) can be highlighted by specifying thresholds. As a consequence, these parameters facilitate identifying anomalies in specific regions of the GPR data.

6. EVALUATION

The presented rendering system was implemented in C++ by extending the one presented by Discher et al. (Discher et al., 2017). Test data consisted of four GPR B-scans continuously captured in driving direction and a 3D point cloud from a mobile mapping LiDAR scan. The four B-scans were captured in parallel, as shown in Figure 7. Two antennas were measuring with a frequency of 2000 MHz, the other two with a frequency of 1000 MHz. Their signals reached a depth of 0.45 m and 0.90 m, respectively. The scanners were mounted with a distance of 0.42 m to each other. Each scanner captured the 650 m of road data with 13 146 data points, holding 512 4-byte samples each.



Figure 7. Ground penetrating radar antennas mounted to the measuring vehicle.

The 3D point cloud is a combination of two LiDAR scans. The respective scanners were oriented to the left and the right, mounted on top of the measuring vehicle. Besides of the three-dimensional coordinates, each measuring point holds information about the measured intensity, visualized as grayscale values. In the region covered by the ground penetrating radar, more than 1 000 points per square meter have been captured.

6.1 Usability

One problem originates from the order of the antennas: While the first and third antenna measured with 2 000 MHz, the second and fourth were using 1 000 MHz. While having these antennas in that order with different frequencies – and therefore different pickup patterns, maximum depths, and accuracies – is useful for having nearly the same region covered with two completely different settings, it makes visualization more challenging, since neighboring B-scans are not directly comparable anymore. Therefore, interpolating the 2D B-scans to create a 3D representation of the captured data might lead to unexpected results when B-scans with differing antenna settings are active. Also, since the B-scans are located close to each other, they easily occlude each other. Since an important part of the visualization is that the coordinates of the radar scan are exactly mapped to the point cloud, adjusting the distance between the B-scans is not a viable option. Thus, the user has to hide unwanted B-scans to make otherwise occluded B-scans visible.

6.2 Performance

As shown by Discher et al., the presented rendering system is generating interactive frame rates for arbitrary large 3D point clouds (Discher et al., 2017). The GPR data is supplied as raw data to our system before being loaded into OpenGL textures to generate the final rendered textures. Therefore, the performance cost in this case is composed of the initial time to load the data, and the time at runtime to update and draw the B-scan textures. Since the few textures in our case have a resolution of $13\,146 \times 512$ pixels with a bit depth of 4 bytes for the raw data and one byte for result textures, and both updating and drawing of these textures is completely GPU-accelerated, this runtime cost is negligible in comparison to the one introduced by managing and rendering the 3D point cloud. However, this was tested on a small data set of 650 m captured road data. For bigger data sets, more advanced memory managing methods and level of detail approaches might be needed in order to overcome challenges such as limited GPU memory.

7. CONCLUSIONS AND FUTURE WORK

A combined visualization of GPR data sets and 3D point clouds enables comparisons between both data sources and, thus, facilitates evaluations (e.g., in the context of road inspections). Anomalies in the GPR data can be compared with the 3D point

cloud to detect irregularities visible from above ground. Manhole covers and gullies can easily be identified within the 3D point cloud and their location can be taken into account when evaluating anomalies from the GPR data. Cropping the GPR B-scans to a certain area of interest in length and height enables focusing on details in a small area and avoids occluding too much context information. Individual B-scans can be enabled or disabled for the visualization to further decrease visual clutter of currently unneeded data, especially with regard to B-scans scanned with different frequencies. As discussed, an implementation of level of detail approaches for the GPR data might improve handling larger data sets.

Visualizing B-scans slightly raised above ground, also raising the ground points from the 3D point cloud in close proximity with them, results in a less occluded view onto the B-scans. By hiding those elevated points in a small region around the cursor, the top plane of the ground penetrating radar cuboid can still be inspected. The threshold and amplification manipulation assist while visually identifying anomalies in the GPR data. An extended automated analysis could help to find anomalous regions in the data by highlighting them during inspection, which might further facilitate finding areas of interest. Another possible improvement is an automated object detection within the 3D point cloud data that would enable the detection of manholes, so anomalies located closely to them could already be respectively tagged in the visualization. Also, the visualization of B-scans from different GPR data sets in areas of intersections and for roads with multiple scanning runs holds potential for further development.

ACKNOWLEDGMENTS

Data provided by “Amt für Geoinformation, Vermessung und Kataster”, City of Essen, Germany.

REFERENCES

- Airfield Inventory, 2018. <http://www.xyht.com/lidarimaging/huge-airport-one-vehicle/>. Accessed: 2018-06-27.
- Benedetto, A., Tosti, F., Ciampoli, L. B. and D’Amico, F., 2017. An overview of ground-penetrating radar signal processing techniques for road inspections. *Signal Processing* 132, pp. 201–209.
- Biasion, A., Bornaz, L. and Rinaudo, F., 2005. Laser scanning applications on disaster management. In: *Geo-information for Disaster Management*, Springer, pp. 19–33.
- Boucheny, C., 2009. Interactive Scientific Visualization of Large Datasets: Towards a Perceptive-Based Approach. PhD thesis, Université Joseph Fourier, Grenoble.
- Boulch, A., Saux, B. L. and Audebert, N., 2017. Unstructured point cloud semantic labeling using deep segmentation networks. In: *Proceedings of 3DOR*, Vol. 2, p. 1.
- Butler, H., Finnegan, D. C., Gadowski, P. J. and Verma, U. K., 2014. plas. io: Open source, browser-based webgl point cloud visualization. In: *AGU Fall Meeting Abstracts*.
- Chen, D., Wang, R. and Peethambaran, J., 2017. Topologically aware building rooftop reconstruction from airborne laser scanning point clouds. *IEEE TGRS* 55(12), pp. 7032–7052.
- Davis, J. L. and Annan, A. P., 1989. Ground-penetrating radar for high-resolution mapping of soil and rock stratigraphy. *Geophysical prospecting* 37(5), pp. 531–551.

- Discher, S., Richter, R. and Döllner, J., 2018. A scalable WebGL-based approach for visualizing massive 3d point clouds using semantics-dependent rendering techniques. In: *Proceedings of Web3D*, pp. 19:1–19:9.
- Discher, S., Richter, R. and Döllner, J., 2017. Interactive and view-dependent see-through lenses for massive 3d point clouds. In: *Advances in 3D Geoinformation*, pp. 49–62.
- Discher, S., Richter, R., Trapp, M. and Döllner, J., 2019. Service-oriented processing and analysis of massive point clouds in geoinformation management. In: *Service-Oriented Mapping*, Springer, pp. 43–61.
- Eitel, J. U., Höfle, B., Vierling, L. A., Abellán, A., Asner, G. P., Deems, J. S., Glennie, C. L., Joerg, P. C., LeWinter, A. L., Magney, T. S. et al., 2016. Beyond 3-d: The new spectrum of lidar applications for earth and ecological sciences. *Remote Sensing of Environment* 186, pp. 372–392.
- Elmqvist, N. and Tsigas, P., 2008. A taxonomy of 3d occlusion management for visualization. *IEEE TVCG* 14(5), pp. 1095–1109.
- Elseberg, J., Borrmann, D. and Nüchter, A., 2013. One billion points in the cloud—an octree for efficient processing of 3d laser scans. *ISPRS P&RS* 76, pp. 76–88.
- Evans, R. D., Frost, M. W., Stonecliffe-Jones, M. and Dixon, N., 2008. A review of pavement assessment using ground penetrating radar (gpr).
- Gao, Z., Nocera, L., Wang, M. and Neumann, U., 2014. Visualizing aerial lidar cities with hierarchical hybrid point-polygon structures. In: *Proceedings of Graphics Interface*, pp. 137–144.
- Geo Radar: 3D and GPR, 2005. <http://www.georadar.pl/en/methods/georadar/3d/>. Accessed: 2018-06-25.
- Giannopoulos, A., 2005. Modelling ground penetrating radar by gprmax. *Construction and building materials* 19(10), pp. 755–762.
- Goswami, P., Erol, F., Mukhi, R., Pajarola, R. and Gobbetti, E., 2013. An efficient multi-resolution framework for high quality interactive rendering of massive point clouds using multi-way kd-trees. *The Visual Computer* 29(1), pp. 69–83.
- Gross, M. and Pfister, H., 2011. *Point-based graphics*. Morgan Kaufmann.
- Huston, D. R., Pelczarski, N. V., Esser, B. and Maser, K. R., 2000. Damage detection in roadways with ground penetrating radar. In: *Eighth International Conference on Ground Penetrating Radar*, Vol. 4084, pp. 91–95.
- Jaakkola, A., Hyyppä, J., Hyyppä, H. and Kukko, A., 2008. Retrieval algorithms for road surface modelling using laser-based mobile mapping. *Sensors* 8(9), pp. 5238–5249.
- Kumar, S., Deshpande, A., Ho, S. S., Ku, J. S. and Sarma, S. E., 2016. Urban street lighting infrastructure monitoring using a mobile sensor platform. *IEEE Sensors Journal* 16(12), pp. 4981–4994.
- Li, F., Elberink, S. O. and Vosselman, G., 2017. Semantic labelling of road furniture in mobile laser scanning data. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*.
- Li, R., 1997. Mobile mapping: An emerging technology for spatial data acquisition. *Photogrammetric Engineering and Remote Sensing* 63(9), pp. 1085–1092.
- Martinez-Rubi, O., de Kleijn, M., Verhoeven, S., Drost, N., Attema, J., van Meersbergen, M., van Nieuwpoort, R., de Hond, R., Dias, E. and Svetachov, P., 2016. Using modular 3d digital earth applications based on point clouds for the study of complex sites. *International Journal of Digital Earth* 9(12), pp. 1135–1152.
- Mobile GPR, 2017. <https://www.catsurveys.com/Services/MGPR>. Accessed: 2018-06-26.
- Olsen, M. J. and Kayen, R., 2013. Post-earthquake and tsunami 3d laser scanning forensic investigations. In: *Forensic Engineering 2012: Gateway to a Safer Tomorrow*, pp. 477–486.
- Preiner, R., Jeschke, S. and Wimmer, M., 2012. Auto splats: Dynamic point cloud visualization on the gpu. In: *Proceedings of EGPGV*, pp. 139–148.
- Puente, I., González-Jorge, H., Martínez-Sánchez, J. and Arias, P., 2013. Review of mobile mapping and surveying technologies. *Measurement* 46(7), pp. 2127–2145.
- Remondino, F., Spera, M. G., Nocerino, E., Menna, F., Nex, F. and Gonizzi-Barsanti, S., 2013. Dense image matching: comparisons and analyses. In: *Digital Heritage International Congress*, pp. 47–54.
- Richter, R., Discher, S. and Döllner, J., 2015. Out-of-core visualization of classified 3d point clouds. In: *3D Geoinformation Science*, Springer, pp. 227–242.
- Rusinkiewicz, S. and Levoy, M., 2000. Qsplat: A multiresolution point rendering system for large meshes. In: *Proceedings of SIGGRAPH*, pp. 343–352.
- Saarenketo, T. and Scullion, T., 1994. Ground penetrating radar applications on roads and highways. Technical report.
- Saarenketo, T. and Scullion, T., 2000. Road evaluation with ground penetrating radar. *Journal of applied geophysics* 43(2-4), pp. 119–138.
- Saito, T. and Takahashi, T., 1990. Comprehensible rendering of 3-d shapes. *ACM SIGGRAPH Computer Graphics* 24, pp. 197–206.
- Schütz, M. and Wimmer, M., 2015a. High-quality point-based rendering using fast single-pass interpolation. In: *Proceedings of Digital Heritage*, pp. 369–372.
- Schütz, M. and Wimmer, M., 2015b. Rendering large point clouds in web browsers. *Proceedings of CESC* pp. 83–90.
- Sigg, S., Fuchs, R., Carnecky, R. and Peikert, R., 2012. Intelligent cutaway illustrations. In: *Proceedings of PacificVis*, pp. 185–192.
- Simons, L., He, S., Tittman, P. and Amenta, N., 2014. Point-based rendering of forest lidar. In: *Proceedings of EnvirVis*, pp. 19–23.
- Teizer, J., Kim, C., Haas, C., Liapi, K. and Caldas, C., 2005. Framework for real-time three-dimensional modeling of infrastructure. *TRR Journal* 1913, pp. 177–186.
- Trapp, M., Glander, T., Buchholz, H. and Döllner, J., 2008. 3d generalization lenses for interactive focus+ context visualization of virtual city models. In: *Proceedings of IV*, pp. 356–361.
- Vaaranemi, M., Freidank, M. and Westermann, R., 2013. Enhancing the visibility of labels in 3d navigation maps. In: *Progress and new trends in 3D geoinformation sciences*, pp. 23–40.
- Zhang, L., Sun, Q. and He, Y., 2014. Splatting lines: an efficient method for illustrating 3d surfaces and volumes. In: *Proceedings of the 18th meeting of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, pp. 135–142.