Constructing Geo-Referenced Virtual City Models from Point Cloud Primitives

Andreas Fricke(✉) and Hartmut Asche

Hasso Plattner Institute, Digital Engineering Faculty, University of Potsdam,
Prof.-Dr.-Helmert-Str. 2-3, 14482 Potsdam, Germany
{andreas.fricke,hartmut.asche}@hpi.uni-potsdam.de

Abstract. This paper presents a novel approach to construct spatially-referenced, multidimensional virtual city models from remotely generated point clouds for areas that lack reliable geographical reference data. A multidimensional point cloud is an unstructured array of single, irregular points in a spatial 3D coordinate system plus time stamp. If geospatial reference points are available, a point cloud is geo-referenced. Geo-referenced point clouds contain a high-precision reference dataset. Point clouds can be utilised in a variety of applications. They are particularly suitable for the representation of surfaces, structures, terrain and objects. Point clouds are used here to generate a virtual 3D city model representing the complex, granular cityscape of Jerusalem and its centre, the Old City. The generation of point clouds is based on two data acquisition methods: active data capture by laser scanning and passive data collection by photogrammetric methods. In our case, very high-resolution stereo imagery in visible light and near infrared bands have been systematically acquired an aerial flight campaign. The spatio-temporal data gathered necessitate further processing to extract the geographical reference and semantic features required in a specific resolution and scale. An insight is given into the processing of an unstructured point cloud to extract and classify the 3D urban fabric and reconstruct its objects. Eventually, customised, precise and up-to-date geographical datasets can be made available for a defined region at a defined resolution and scale.

Keywords: Point cloud · Object reconstruction · Data acquisition

1 Introduction

It is a fundamental human need to appropriate one's immediate environment. A prerequisite to do so is positioning and orientation in space. The term space here refers both to physical reality and to virtual, i.e. modelled and generalised spaces, which either depict abstracted reality or create an artificial cyberspace. Orientation and positioning as well as direction and movement in space requires data gathered in one's environment to establish a spatial reference. Such data can, e.g., be landmarks, stars and physical or temporal distance measurements between them and one's position. It is obvious that in addition to the classical

O. Gervasi et al. (Eds.): ICCSA 2020, LNCS 12252, pp. 448–462, 2020.
https://doi.org/10.1007/978-3-030-58811-3_33

spatial reference, time reference of geospatial data is equally important. This applies not only to everyday applications but also to scientific work in modern geographic information science [11]. Spatial data with explicit temporal reference are therefore referred to as spatio-temporal data. Without exception, geographic data have one or more temporal reference. This can, e.g., either be the acquisition, storage or processing time. Because of these different time stamps, the temporal validity of geospatial data can be assessed. Focusing on the acquisition date, the majority of geodata are considered static and are processed accordingly. This contrasts markedly with the constant dynamics of our natural environment. Hence the need to update geographical data has to be considered when it comes to processing the data for a topical or near real-time application [4,5].

Today, establishing a valid spatio-temporal reference is no longer a specialis's task. Anyone using a mobile device, such as a smartphone or GPS watch, is able to acquire geodata in the form of, e.g., a geocoded photo or GNSS track. Open-source geo databases like OpenStreetMap (OSM) database are compiled and updated by using such crowd-sourced geodata largely collected by volunteers with their smartphones. Thanks to systems like OSM geodata of good to sufficient quality are now globally available. However their regional coverage, scale, precision and topicality varies considerably according to the numbers and efforts of the data acquisition volunteers. Official geodata can compensate that disadvantage only to a certain degree due to access restrictions, the geopolitical environment or outdated data. As a consequence, it remains inevitable to collect up-to-date geodata of a specific area for specific application purposes. Starting from the 1960s, the method of choice for the acquisition of precise, scalable geodata for global, regional and local area coverages is remote sensing from aircraft, satellite and, lately, drone platforms. A variety of sensor technologies is employed for data capture, staring from high resolution aerial survey cameras to multi-spectral sensors, radar and lidar sensors. The spatio-temporal data gathered by any of these systems necessitate further processing to extract the geographical reference and semantic features required in a specific resolution and scale [14]. Eventually, customised, precise and up-to-date geographical datasets can be made available for a defined region at a defined resolution (and scale).

This article presents an innovative approach to construct spatially-referenced, multidimensional virtual city models from photogrammetrically generated point clouds for areas that lack reliable geographical reference data. Here we focus on the processing of the unstructured point cloud to extract and classify the urban fabric and its objects. The work presented here is part of a wider international R+D project on a 3D geovirtual decision support system for community development in East Jerusalem.

2 Use Case: The Complex Cityscape of Jerusalem

The centre of the urban agglomeration of Jerusalem is the ancient, walled Old City. Prototypical of an Oriental City, the East Jerusalem cityscape is characterised by a complex, granular buildup of houses of various dimensions and heights and an irregular network of streets, paths and lanes (see Fig. 2). Buildings usually house multiple functions, e.g. commercial and residential use. The 3D image depicting part of the Old City represents a very high resolution point cloud providing a pseudo-realistic visualisation of the area (see Fig. 1).

2.1 Lacking Geospatial Database Due to Complex Geopolitical Situation

The agglomeration of Jerusalem, and East Jerusalem in particular, is an area where precise and reliable geotopographic data are not easily available or accessible, respectively. This is a result of an ongoing complex geopolitical situation in which Israel has annexed East Jerusalem to its territory contrary to international law. Crowd-sourced geodata, such as the OSM database, cannot compensate for the lack of official survey data, since the OSM system does not enforce a strict, authoritative geometric and semantic quality control of the data largely gathered by non-specialist volunteers [4,5]. In addition, the cityscape of Jerusalem is exposed to very high dynamics in the built-up area and infrastructure development. As a consequence, taking account of all relevant circumstances, a realistic, practicable way to access and process high-resolution, up-to-date geodata of East Jerusalem is the acquisition by an airborne multi-spectral digital camera system [15]. The data stock used in the project presented here has been collected in autumn 2019.

2.2 Applications

The geospatial data stock forms a uniform, topical high-resolution basis to construct a virtual 3D city model of East Jerusalem dedicated to the web-based use by the civil society and its institutions in the area. The virtual city model will come with easy-to-use GIS functionalities, facilitating a range of spatio-temporal applications, such as information and documentation, urban land use and land management, education and health infrastructure, technical infrastructure, tourism, planning purposes and decision making.

3 Approach

This paper deals with a novel approach to construct a geo-referenced virtual city model of East Jerusalem from unorganised, unstructured point clouds. As we deal with a granular cityscape, a particular focus is on the classification and extraction of building models [6,7,10]. The novelty of our approach is the fact that we adapt the processing of invariant point data in terms of spatio-temporal reference to the construction of multidimensional city models. In that way, a generic processing (and acquisition) technique is made available for the construction of both geospatial cityscapes and landscapes in areas where no reliable reference data are available. During the construction process, an elastic grid is generated by means of voxels in different levels of resolution and hierarchy,

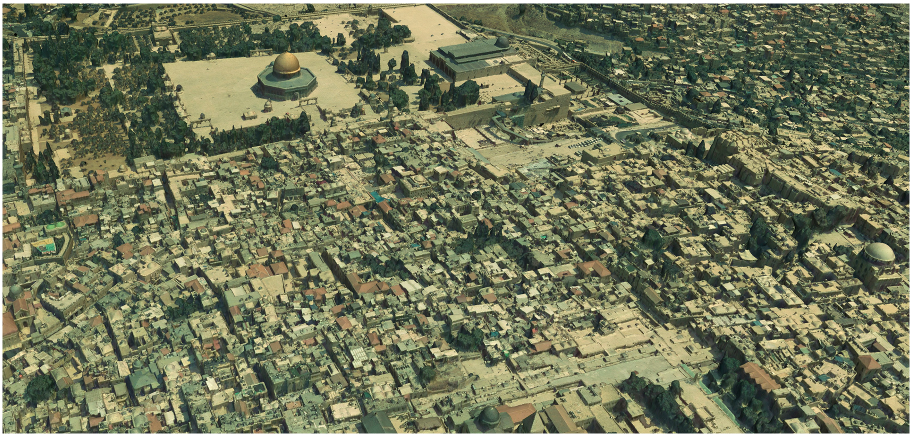

Fig. 1. Old City of East Jerusalem: overview, point cloud representation

which represents and maps the point cloud [2,17]. Applying established methods of computer graphics, object surfaces are then projected and constructed using iterative resampling methods [1,13]. Particular attention is paid to the point consolidation process, with respect to uniform point distribution as well sufficient sampling density at minimum distance from point to voxel. In addition, the requirements point clouds need to comply with in order to perform the process in multidimensional point clouds, are checked during pre-processing [16]. The solution approach outlined above is developed within the scope of a wider R+D effort that deals with the with the creation, (re-)construction, administration and maintenance of virtual 3D city models in constantly changing urban environments [4,5]. An essential prerequisite for this work is the volume and quality of the geospatial source data. The more precise the source data record or map geographical reality, i.e. spatio-temporal object features or geometric and multispectral resolution, the higher the degree of the pseudo-realistic visualisation of the resulting city model can be [12].

3.1 Point Clouds as a Comprehensive Basic Data Set

To date, remotely sensed point clouds can best meet the above requirements. Point clouds can also easily capture the spatio-temporal dynamics typical of urban environments. Point clouds easily map 3D surfaces and objects by a densely spaced sequence of points storing the detected geometry of objects and surfaces point-wise. Thus the totality of points, the point cloud, is a versatile geospatial data representation of complex terrestrial surfaces, such as cityscapes [12]. Because of their geometric nature, point clouds are particularly well-suited for the derivation and subsequent visualisation of geospatial phenomena and artefacts. Other than classical vector and raster data, point clouds facilitate the extraction of a regional, spatially invariant, geometric base structure [3,14].

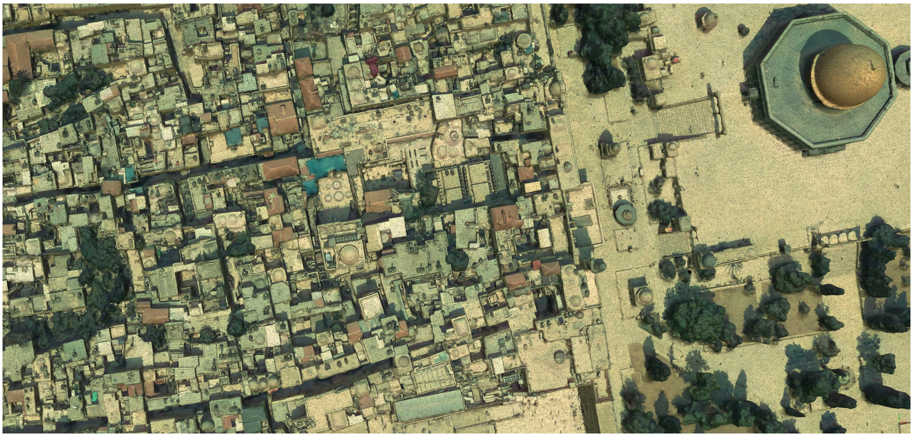

Fig. 2. Old City of East Jerusalem: spatial structures, point cloud representation

Any processing of point cloud data requires to solve a twofold problem: How can addressable objects from any kind of geodata be connected in a meaningful way, and how can these objects be analysed for their potential added value? Solving these R+D issues is considered a major challenge in the interdisciplinary field of spatial sciences [7,11]. Based on the lowest common denominator, the space-time reference, two possible solutions can be identified to model high to very-high resolution 3D geo-objects: schematisation and reconstruction.

Schematisation. This research field deals with the procedures required to schematise available geodata. To link existing data with or without a spatial reference a scheme is essential to do this in a meaningful way. A common scheme is therefore mandatory to facilitate successful harmonisation and fusion of data [14]. Schematisation of different input data is hampered by the data reference which is both spatial and semantic. When source data have different spatial as well as semantic references, which generally is the case, schematisation may result in a loss of reference accuracy. In addition, data schematisation, like, e.g., interpolation, is a one-way process. As a result, the original source data are irreversibly altered. This is compensated by the fact that geospatial data different sources can successfully be processed jointly and stored in uniform datasets. Today, structured schematisation, harmonisation and integration of different data forms and formats is implemented in the quasi-standard Feature Manipulation Engine (FME) or in the INSPIRE Guidelines of the EU. In contrast, schematisation of references is a somewhat disregarded R+D field as it directly affects the accuracy of geodata.

Reconstruction. This research field investigates the use of computational engineering methods to approximate an artefact or semantics by constructing both virtually. Hence, an artificial reconstruction of the original data is carried out without directly manipulating them, thus creating new data while leaving the source data unaltered. This is the rationale behind machine learning and artificial intelligence in general. Although both schematisation and reconstruction are roughly based on the same principle, they differ in their handling of source data. This is apparent when it comes to the application of 3D building models which have massively increased in importance far beyond simple visualisations in times of so-called digital twins [5]. One driving factor behind this development is the rapidly growing availability of high-resolution input data. What is lacking, however, is an executable generic process applicable to all current domains and solutions. For the time being, only domain-specific approaches can be found. Consequently, the level of abstraction is currently limited to the domain and therefore increasingly schematised.

3.2 Acquisition of Uniform Geodata Base

It has been pointed out that the availability and quality of source data is an essential prerequisite for the construction of a virtual 3D city model of the Jerusalem agglomeration. The ideal database would be a uniform data with identical spatial resolution and parametrisation that covers the entire study area [5]. As has been mentioned, such spatio-temporal data are not readily available, neither from topical official survey data nor from crowd-sourced OSM data or other regional data pools. There is also a lack of high-resolution regional reference data. To generate a uniform high-resolution geo database as the single source used throughout the R+D project, an aerial flight campaign was commissioned to systematically acquire very high-resolution stereo imagery in visible light and near infrared bands. Figure 3 illustrates the flight campaign of September 2019 with flight strips and study area. In the area where flight strips are orthogonally overlaid the Old City of Jerusalem can be found. It goes without saying that the acquired database bears a time stamp of the collection time. All of its data represent the real-world status at the time of acquisition. This is important to note, since cityscapes such as Jerusalem are subjected to permanent spatial change over time. However, such urban dynamics are not mirrored in the available static data pool [7]. The time stamp attached to the geo-objects captured serves as an independent, invariant variable specific to each object. Each new recording of the same object at a particular time comes with a new time stamp making it possible to distinguish the same object over different points in time and trace its spatio-temporal development. The following time is relevant to geospatial databases: Real time refers to a fully synchronised digital replica of present reality with events, people, places and spatial processes, future time relates to modelling, simulation or projection of future spatial scenarios, and past time is used for the documentation and interpretation of historical events, places and processes. To keep an acquired geospatial database up-to-date and usable over time, updating of the data either at defined points in time or permanently is essential.

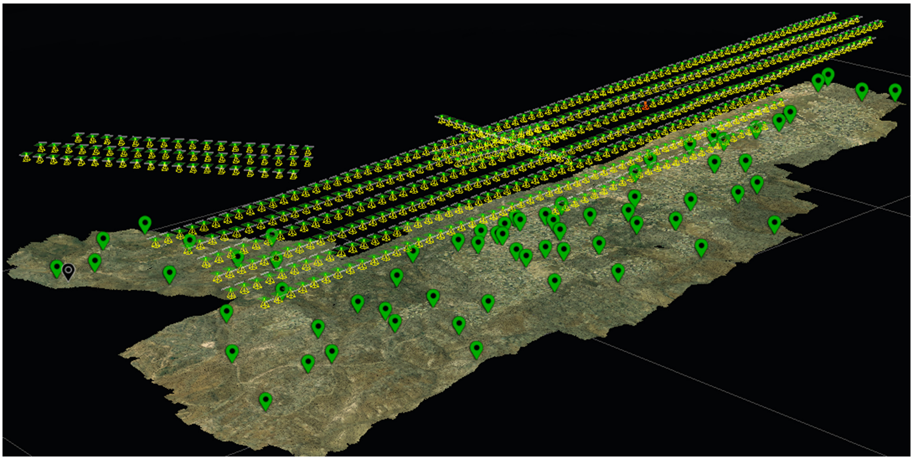

Fig. 3. Overview of flight campaigns and study area East Jerusalem; source Bentley ContextCapture (v15)

4 Generation of Geo-Referenced Point Clouds

Characteristics. A multidimensional point cloud is a 3D or 4D data set of a single, irregular and unstructured point array in a spatial 3D coordinate system defined by x, y, z coordinates and the time component [16]. The result is a registered point cloud. If reference points are available, a point cloud is geo-referenced. Assuming a spatial reference, point clouds contain an high-precision reference dataset, since every single point has a very high position accuracy and fidelity compared with classical vector and raster spatial models. Therefore, referenced point clouds are often used for secondary referencing of other geodata. Point clouds can be employed in a variety of applications. They are particularly suitable for the representation of surfaces, structures, terrain and objects [15]. Point clouds are used for documentation purposes or further processing in, e.g., CAD, BIM or in 3D rendering software for 3D modelling or 3D visualisation [12].

Generation. The generation of primary point clouds is based on two data acquisition methods: Active data capture by laser scanning and passive data collection by photogrammetric methods [10]. 3D laser scanning is based on the principle of beam tracing or distance measurement using a laser beam. The scanner emits laser pulses at extremely short intervals. The pulsed photons have a high energy. They are reflected by the objects targeted, and the returned beams are detected again in the transmitter. Based on the speed of light, from the transit time measurement the object distance can be calculated. Similarly, the spatial coordinates of a targeted point, relative to the scanner position, can easily be calculated, since horizontal and elevation angles are implicitly measured. Photogrammetric point clouds, in contrast, are based on the image-based evaluation of co-existing object-image points in at least two different image data. Their accuracy depends on the inner orientation (camera calibration) and outer orientation (camera orientation), as well on as potential reference points. Corresponding image points are mathematically mapped using a bundle block adjustment. This method to calculate 3D points is known as stereo-photogrammetric processing. It allows to calculate all unknown parameters of the collinearity equations. In addition, an near-reality colour value can be extracted by means of photogrammetric processing. Active laser scanning allow the registration of one coded colour value only per point [16].

4.1 Data Acquisition, Database

Implementing the approach detailed above, aerial imagery obtained from a flight campaign is evaluated photogrammetrically to generate a point cloud for the study area of East Jerusalem. A total of 571 RGBI (red, green, blue, near infrared) nadir aerial images are available acquired in two flight campaigns of 23.09.2019 with a ground resolution of (minimum) 10 cm GSD, taken with a camera system of the Z/I DMC 250 II e type (see Fig. 3). The first campaign covers the entire investigation area of East Jerusalem with 525 images (forward overlap 80%, side overlap 60%). The imagery covers an area of nearly 190 km^2, corresponding to a data volume of almost 135 gigapixels. The second campaign covers the Old City of Jerusalem only with 46 images (forward overlap 80%, side overlap 80%) encompassing an area of nearly 13.5 square kilometres, which corresponds to a data volume of almost 11 gigapixels. This campaign is located within the area covered by the first Campaign. Corresponding pixels have been recorded in at least 10 different aerial images in the entire study area, with exceptions in the peripheral areas. In the area of the Old City, corresponding pixels are present in at least 30 different images due to the overlay of the flight campaigns. A total of 70 ground reference points (differential GPS), equally distributed across the study area, are available for geo-reference. All aerial images are referenced by GPS and INS. Bentley's ContextCapture (v14) software system is used to perform referencing of the input imagery as well as of 69 out of 70 ground points, and is also employed for bundle block adjustment (see Fig. 3). Essential parameters are the collinearity equations at pixel level and to maximally increase the quality of matching. The processing is performed on a mini-cluster consisting of two identical workstations. An AMD Ryzen 9 3900X 12-core processor, 64 GB, NVMe memory and NVIDIA GeForce RTX 2080Ti are used for the calculation. The mini-cluster allows, among other things, parallel computation and simultaneous use of several graphics cards. It shows that a very consistent dataset or bundle block is provided, since the global error with ground reference points is about half a pixel; the median is even slightly below. Turning to the point clouds generated, it can be seen that these cover massive volume of data and, with more than 100 points per square meter, represent the full pixel density of the source data. Note that this number refers to a 2D reference surface only. However, it includes all points of the surface, i.e. the 3D points, too. also. The theoretical maximum for photogrammetric processing in the reference plane X & Y is 100 points per square meter (image data 10 cm GSD), or 91 points in the computed block.

4.2 Structuring of Point Clouds

Within the framework described above, 3D respectively 4D point clouds are photogrammetrically generated in the pre-processing. Subsequently, the data space is subdivided by hierarchical methods, such as octree or n-tree. Octree, for example, defines a hierarchy that can be used to inspect large point clouds and interact with them with high performance [12,14]. An octree is a data structure for indexing three-dimensional data. It extends the concept of binary trees and quadtrees, which structure 1D or 2D data. Each node of an octree represents a volume as a cuboid. Furthermore, these cubes are often aligned with axes of the coordinate system. Each octree node has up to eight subnodes. If a node has no subnodes, the corresponding cube can be represented uniformly and no further subdivision is necessary. Each node of the spatial system with associated subnodes is characterised by the presence or absence of points. In the representation of volumetric data, such as building objects, a further subdivision of a node is not necessary if the mapped volume is completely uniform and a reference plane is known. Nodes that have points will become part of a spatially adaptive, elastic hierarchical voxel grid in the further process. Elasticity includes the possibility to use different resolutions in a grid in an advantageous way.

Volume graphics is a different method of modeling as a voxel data set [1]. Here a value is recorded at a single point from the object at regular intervals. The result is a cube-shaped 3D grid of voxels. These voxel models can easily be converted into images, which, depending on the minimum resolution, can be extremely pseudo-realistic and detailed [2]. This type of modelling requires significantly higher resources in direct comparison to triangle-based polygonal meshes and has a performance loss in visualisation compared to a polygonal surface model of similar size. In addition, the manipulation of these data is much more complex [1]. The ultimale goal of all algorithms briefly touched on here is to model 3D objects. This is usually done, even on the most modern graphics hardware, by means of simple triangular structures, which in mass reproduce and represent complex polygonal surfaces [13]. Also, despite the inclusion of complex 3D textures in the volume graphics, the hardware acceleration is still focused on triangle computation. Hence a combination of both approaches proves to be expedient, as the example of the Marching Cubes algorithm from 1987 in imaging medicine proves [9]. For the first time it was possible to approximate inefficient volume models by efficient polygonal surface models, to visualise them effectively and to combine the advantages of both approaches [2].

4.3 Generation of Urban 3D Objects Using the Marching Cube Algorithm

The core idea of Marching Cubes is to decompose a given voxel model of an object into small cubes and then march from one cube to the next and determine how the surface of the object intersects the respective cube [9]. A selected threshold value regulates which parts of the relevant cube lie inside or outside the object. The distinction between solid and transparent is fundamental to the procedure. It affects the normal calculation of the surface, since the slope of the scalar field at each voxel point is also the normal vector of a hypothetical iso surface running from that point. There are 256 possibilities of how an arbitrarily shaped surface can divide a voxel into interior and exterior areas, as, based on combinatorics, there are 2 to the power of 8 possibilities to divide the eight corners of a voxel into two disjoint sets inside and outside [9]. Due to symmetry effects, however, the number is reduced to only 15 different variants. The so-called Triangle Lookup Table contains all these possibilities. The runtime of the classical marching cube algorithm depends significantly on the number of voxels considered. An optimisation is based on the hierarchical investigation beforehand, as described, to use only voxels as input that contain points and thus represent an object [8,17].

It emerges that the Marching Cube algorithm represents an effective technique for the calculation of iso-surface and object modelling in 3D [17]. Consequently, a combination of volume graphics and triangle-based polygonal meshes based on a well-established algorithm can be used to approximate and reconstruct the surface and thus the hull of an object [8]. Finally, this geometry requires to be characterised and extracted as an object in order to enrich it with spatially referenced semantics for use in a spatial information system of the entire study area.

The essential work steps in this context include a meaningful subdivision of a point cloud [16]. This has to be manageable for algorithms working on it. The subdivision can be based on either the data structure, as described, using a hierarchical tree structure, or with spatial or structural filtering within the process. Similar to a segmentation, this reduction represents an adjustment factor, since an object can occur in several areas, nodes or tiles [14]. An overlapping of areas to be processed also needs to be considered so that no break between regions occurs. Similarly, it has to be made sure that an object has identical characteristics across regions. In other words, voxels or horizontal slices of a point cloud, as familiar from imaging medicine, are fed into an algorithm [2]. The algorithm reconstructs an object from them bit by bit, as one would do with, say, with a finite number of small Lego bricks. Depending on the resolution, i.e. how high the layers and the voxel are, an object image, for example of a building, is gradually created. If the process is repeated and parameters of resolution are changed, there is also an incremental approach that can model an object in its entirety or in a spatio-temporal manner. This results is a volume model of an object as well as a polygonal triangular model that can be stored in a database. Depending on the process, different levels of detail are conceivable for one and the same object.

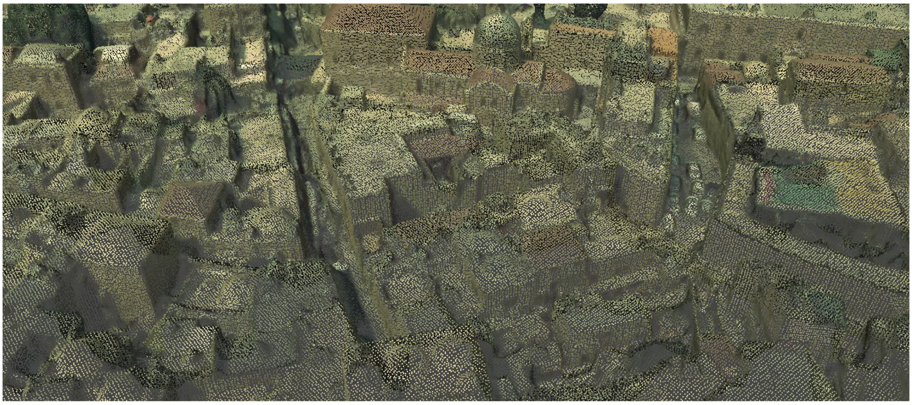

Fig. 4. Point cloud (point spacing 30 30 cm)

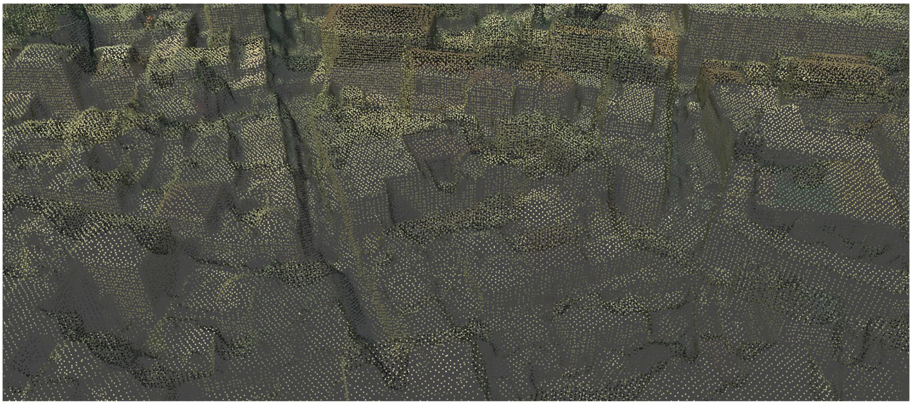

Fig. 5. Point cloud (point spacing 50 50 cm)

5 Implementation Issues

In a nutshell, the methodological concept of this approach addresses the research question of how accurately and precisely multidimensional virtual spatial models correspond to the complex reality they present. It challenges the widespread approach to different virtual models by different levels of detail (LOD) [14]. Since the availability of high-resolution output data such as point clouds has increased significantly, it can be assumed that the concept of static LOD degrees is less appropriate to represent very-high resolution complex objects. In addition, discretisation of the source data is always applied when preparing a LOD inventory. It is important to consider the geometric tolerances of existing LODs with regard to quality assurance. In this context, discretisation of the source data to known LODs is possibile but not mandatory. Even without discretisation, the full data depth available can be utilised. The above issues need further analysis and evaluation. Tests have shown that the approach outlined here includes some open issues that need to be addressed. Figure 6 shows derived and extracted two-dimensional rings generated with a provisional version of the layer algorithm. It is important to note that this layer is based on the terrain's reference plane only. In the area of each outline, mean heights for the individual object areas can be derived from the point cloud and added to the data as a semantic feature. As shown in Fig. 7, the result is a 2.5 D data set containing extruded building outlines, which can be semantically augmented with further attributes. Both Figs. 6 and 7 show the current status quo of point cloud processing.

Another issue to be addressed when implementing the algorithm is the reduction of objects to two dimensions. Without exact segmentation and classification of a 2D slice of a point cloud's reference plane, it is sometimes problematic to derive the exact extent and position of an outline generated [6]. This may result in a wrong rotation or over-extraction of an object. A point cloud can also have

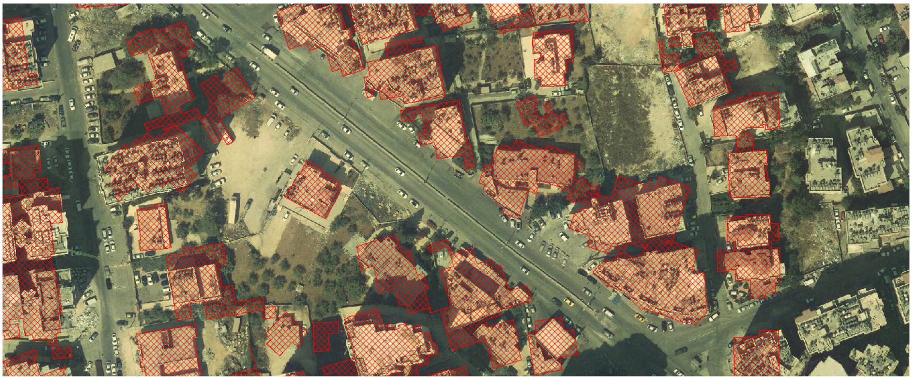

Fig. 6. 2D building outlines from point clouds

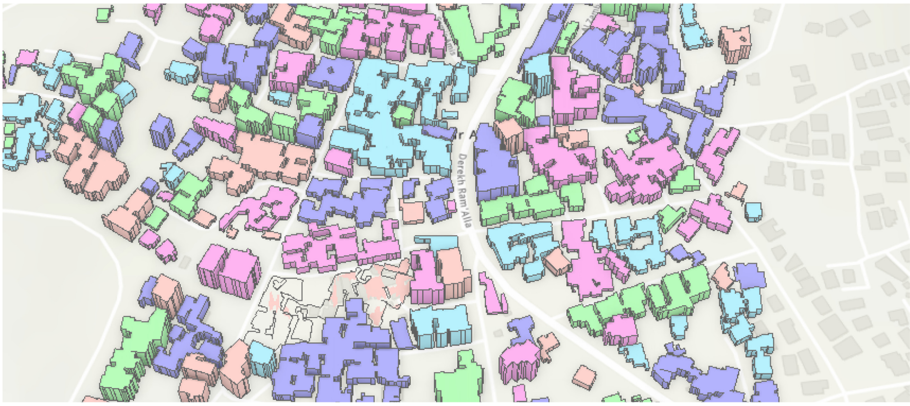

Fig. 7. Extruded building outlines in 2.5D

a too high resolution for an algorithm (see Figs. 2 and 1). Likewise, photogrammetric point clouds of building facade elements may contain few or no points due to missing corresponding pixels. Contrary to that, the Marching Cube algorithm requires a consistent database. Hence, no values can be generated where no values are available. At this point, one option is the reconstruction of buildings by means of voxels. Buildings usually have a cubic or cuboid shape, hence it can be assumed that facades can be reconstructed in a simple geometric way (see Fig. 5). In contrast, if an object is represented in a point cloud of a very high resolution, and this resolution is not required to represent this object distinctly, resources are wasted (see in contrast Figs. 1 and 5). It is therefore a matter of the correct dimension and an incremental and interactive approach to extract the most realistic object from the point cloud. Potential applications are, for example, the analysis of the deviation between vector, raster and point cloud representations that map a building object. Differences can reveal geometric accuracies between the different representations, but also show structural changes. Typically, objects are analysed by algorithms, especially in machine learning, with the help of similarities, and are successively derived and constructed. Consequently, a building model is broken down into its components (i.e. walls and roof) and then, by means of trained similarities, it is determined which part of the object is to be represented and how [3,6,10]. This requires a substantial number of potential object components that can be compared to automatically derive the required object component. Overall, the mathematical morphology of point clouds is highly suitable to be processed in this context, since the most versatile objects from a comprehensive, invariant 4D data set are available in a complete and simple form. With machine learning methods, as with the marching cube algorithm, considerable added value can be gained from geodata with an ever-present link to the original data.

6 Conclusion

This article presents an innovative approach to construct spatially referenced multidimensional virtual city models from photogrammetric point clouds for areas where no reliable geographic reference data are available. The work presented here is part of a larger international R+D project that investigates the generation, (re-)construction, management and maintenance of virtual 3D city models in a constantly changing urban environment. To date, the project work carried out has shown that the generation and processing of point clouds can provide quality geodata to create and maintain a uniform, fully referenced 3D geodata base in a limited period of time, if not on an ad-hoc basis. In the study area of East Jerusalem, the lack of precise, up-to-data geospatial data necessitates the airborne acquisition of geospatial point data conforming to the key requirements of resolution, data quality and topicality. The resulting high-resolution point cloud can be used for a variety of applications. One relevant usage is the extraction of virtual building models from high-resolution point clouds. The use of point clouds is of crucial importance in East Jerusalem, where no primary data source is able to provide geospatial data representing the complex urban reality and its spatio-temporal dynamics. Because of their purely geometric content, point clouds are especially suitable for the derivation and visualisation of geospatial phenomena and artefacts. To extract building models from point clouds, a procedure originating from the field of imaging medicine is adapted, allowing for the modelling of 3D objects in (very) high resolution. Drawing on methods of volume graphics (voxel grids) and classical computer graphics (polygonal triangular meshes), this novel approach facilitates the effective processing of point clouds. In that context, the structuring of the source data is given special attention. Preliminary applications of the approach discussed here prove that extruded building outlines can successfully be extracted and visualised.

Acknowledgements. The work discussed here is part of a larger R+D project on East Jerusalem with Palestinian, East Jerusalem, and NGO partners funded by the European Union. Part of this research work is supported by a PhD grant from the HPI Research School for Service-Oriented Systems Engineering at the Hasso Plattner Institute for Digital Engineering, University of Potsdam. The funding of both institutions is gratefully acknowledged.

References

1. Akenine-Möller, T., Haines, E., Hoffman, N.: Real-Time Rendering. CRC Press, Boca Raton (2019)
2. Ashburner, J., Friston, K.J.: Why voxel-based morphometry should be used. Neuroimage **14**(6), 1238–1243 (2001)
3. Brenner, C.: Building reconstruction from images and laser scanning. Int. J. Appl. Earth Obs. Geoinf. **6**(3–4), 187–198 (2005)
4. Fricke, A., Döllner, J., Asche, H.: Servicification – trend or paradigm shift in geospatial data processing? In: Gervasi, O., et al. (eds.) ICCSA 2018. LNCS, vol. 10962, pp. 339–350. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-95168-3_23
5. Fricke, A., Asche, H.: Geospatial database for the generation of multidimensional virtual city models dedicated to urban analysis and decision-making. In: Misra, S., et al. (eds.) ICCSA 2019. LNCS, vol. 11621, pp. 711–726. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-24302-9_52
6. Haala, N., Kada, M.: An update on automatic 3D building reconstruction. ISPRS J. Photogrammetry Remote Sens. **65**(6), 570–580 (2010)
7. Hall, T., Barrett, H.: Urban Geography. Routledge, Abingdon (2018)
8. Huang, M., Wei, P., Liu, X.: An efficient encoding voxel-based segmentation (EVBS) algorithm based on fast adjacent voxel search for point cloud plane segmentation. Remote Sens. **11**(23), 2727 (2019)
9. Lorensen, W.E., Cline, H.E.: Marching cubes: a high resolution 3D surface construction algorithm. ACM Siggraph Comput. Graph. **21**(4), 163–169 (1987)
10. Mayer, H.: Object extraction in photogrammetric computer vision. ISPRS J. Photogrammetry Remote Sens. **63**(2), 213–222 (2008)
11. Seymenov, K.: Climate elasticity of annual streamflow in Northwest Bulgaria. In: Nedkov, S., et al. (eds.) Smart Geography. KCG, pp. 105–115. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-28191-5_9

12. Richter, R., Discher, S., Döllner, J.: Out-of-ccore visualization of classified 3D point clouds. In: Breunig, M., Al-Doori, M., Butwilowski, E., Kuper, P., Benner, J., Haefele, K. (eds.) 3D Geoinformation Science. LNGC, pp. 227–242. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-12181-9_14
13. Szeliski, R.: Computer Vision: Algorithms and Applications. Springer, London (2010). https://doi.org/10.1007/978-1-84882-935-0
14. Tolpekin, V., Stein, A.: The Core of GIScience: A Process-Based Approach. University of Twente, Faculty of Geo-Information Science and Earth Observation (ITC) (2012)
15. Vosselman, G., Dijkman, S.: 3D building model reconstruction from point clouds and ground plans. Int. Arch. Photogrammetry Remote Sens. Spat. Inf. Sci. **34**(3/W4), 37–44 (2001)
16. Weinmann, M.: Preliminaries of 3D point cloud processing. In: Reconstruction and Analysis of 3D Scenes, pp. 17–38. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-29246-5_2
17. Zhou, Y., Tuzel, O.: Voxelnet: end-to-end learning for point cloud based 3d object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4490–4499 (2018)