

INTEGRATION OF 3D OBJECTS AND TERRAIN FOR 3D MODELLING SUPPORTING THE DIGITAL TWIN

J. Yan^{1,*}, S. Zlatanova¹, M. Aleksandrov¹, A. A. Diakite¹, C. Pettit¹

¹ UNSW Built Environment, Red Centre Building, Kensington NSW 2052, Sydney, Australia
(jinjin.yan, s.zlatanova, mitko.aleksandrov, a.diakite, c.pettit)@unsw.edu.au

Commission IV, WG IV/10

KEY WORDS: 3D Objects, Terrain, Integration, 3D Modelling, Digital Twins

ABSTRACT:

3D modelling of precincts and cities has significantly advanced in the last decades, as we move towards the concept of the Digital Twin. Many 3D city models have been created but a large portion of them neglect representing terrain and buildings accurately. Very often the surface is either considered planar or is not represented. On the other hand, many Digital Terrain Models (DTM) have been created as 2.5D triangular irregular networks (TIN) or grids for different applications such as water management, sign of view or shadow computation, tourism, land planning, telecommunication, military operations and communications. 3D city models need to represent both the 3D objects and terrain in one consistent model, but still many challenges remain. A critical issue when integrating 3D objects and terrain is the identification of the valid intersection between 2.5D terrain and 3D objects. Commonly, 3D objects may partially float over or sink into the terrain; the depth of the underground parts might not be known; or the accuracy of data sets might be different. This paper discusses some of these issues and presents an approach for a consistent 3D reconstruction of LOD1 models on the basis of 3D point clouds, DTM, and 2D footprints of buildings. Such models are largely used for urban planning, city analytics or environmental analysis. The proposed method can be easily extended for higher LODs or BIM models.

1. INTRODUCTION

The validity of digital models is important for supporting planners and decision-makers in exploring the built environment (Pettit et al., 2012). Modelling buildings, precincts and cities in 3D provides a better perception and more elaborated options for supporting city analytics planning, decision-making and community engagement (Zlatanova, 2000). However, as we embark on the journey of creating Digital Twin representations of our cities as briefly outlined by (Batty, 2018), we need to solve the fundamental integration challenges associated with 3D models and terrain.

The construction of the 3D models has developed very rapidly over the past few decades. 3D models can be created from airborne laser scanning data (Haala , Brenner, 1997, Verma et al., 2006), 2D GIS data (Shiode, 2000, Ledoux , Meijers, 2009), aerial photographs and terrestrial laser scans (Fruh , Zakhor, 2001), or from UAV images (Xie et al., 2012), etc. These reconstruction procedures are focused either on buildings or on the bare terrain. As a result, terrain and 3D objects are maintained independently and integrated only for specific purposes.

A 3D urban model should contain not only the 3D objects, but also 3D terrain (especially for hilly places), because buildings are designed and constructed on it (Li et al., 2004). Often, the reconstructed 3D objects (e.g., buildings) are assumed standing on a plane surface, and this surface is regarded as the default terrain. In fact, the natural terrain is irregular and uneven, rather than planar. Thus, when integrating 3D city objects with the terrain, a topological issue will occur, that is the 3D objects may float over or sink into the terrain.

To address this issue, the City Geography Markup Language (CityGML) (Gröger et al., 2008) introduced *TerrainIntersectionCurve* (TIC), which is a curve that indicates where 3D objects are touching the terrain (Figure 1). Then, the terrain is locally fixed to fit the TIC. By this means, the TIC also ensures the correct positioning of textures or the matching of object textures with the DTM. Since the intersection with the terrain may differ depending on the LOD, a 3D object may have different *TerrainIntersectionCurves* for all LOD.



Figure 1. *TerrainIntersectionCurves* in CityGML (Gröger et al., 2008).

Therefore, the key point is to obtain the correct TIC, and amend 3D objects and the terrain appropriately based on TIC.

Before discussing the approaches for computing TIC, we should clarify the possible source data as they can be available in different forms and have influence on the procedure. 3D objects can be created by extruding footprints, using pre-reconstructed 3D models or as BIM models. The terrain can be represented as point clouds, TIN or grid. Therefore, we distinguished five possible approaches based on these

*Corresponding author

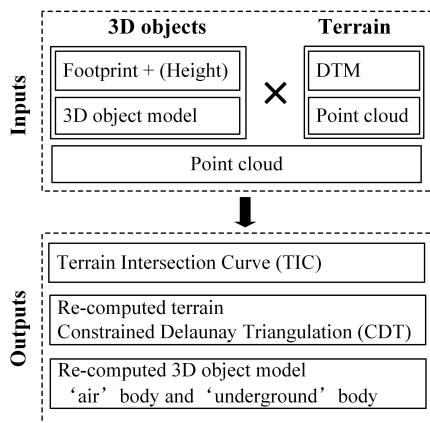


Figure 2. The inputs and outputs of this research.

different inputs (Figure 2): (i) footprints (with information about building height) + DTM, (ii) footprints + Point cloud, (iii) 3D objects + DTM, (iv) 3D objects + 3D point clouds, and (v) Point cloud only.

In this paper, we elaborate on methods to compute TIC based on the five different combinations of source data sets and taking buildings as 3D objects. In this research one approach is implemented and evaluated. The paper is organised as follows: Section 2 explains the fundamental concepts underpinning the research and Section 3 presents the five possible approaches based on different input combinations. Discussion on different approaches is made in Section 4. Section 5 demonstrates and evaluates the approach using the UNSW campus precinct in Sydney and Section 6 concludes by outlining future work.

2. FUNDAMENTAL CONCEPTS

2.1 Footprints

The notion of footprint is mostly referred to buildings. Nonetheless, other city objects also have footprints, such as streets and green areas (Beil , Kolbe, 2017). Building footprint is the area used by the building structure, which is defined by the perimeter of the building plan¹. Footprints of buildings can be found in a large number of different data sets, e.g., Open Street Map (OSM) (Fan et al., 2014), can be reconstructed from airborne LiDAR data (Zhang et al., 2006, Hammoudi et al., 2009), or from digital surface models (Brédif et al., 2013). The building footprint is fundamental for a number of urban precinct modelling tools such as the Envision Scenarios Planner (Trubka et al., 2016).

2.2 Height of buildings

Building height can be defined as the average maximum vertical height of a building or structure measured at a minimum of three equidistant points from finished grade to the highest point on the building or structure along each building elevation. Architectural elements such as parapet walls, chimneys, vents, and roof equipment are not considered as part of the height of a building or structure². Only the selection process of the starting point is slightly different, for instance, the starting point is

chosen based on the difference between the highest and lowest foundation point³. In Figure 3, the *A* and *B* are the highest and lowest foundation points respectively, *C* and *E* are on the same plane with *A*, *F* is the mid-point of foundation, and *D* is the highest point of the building. If there is less than 3 metres elevation change between points *A* and *B*, the measure from point *A*, but if there is a 3 metres or greater elevation change between *A* and *B*, measure height from point *B*.

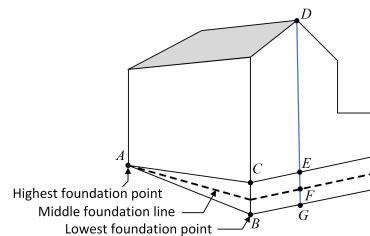


Figure 3. The rule of measuring building height.

In this paper, we consider a building height measured from the mid-point (*F*) of the foundation to the highest point *D*, no matter if the elevation changes between *A* and *B* for more or less than 3 metres. Therefore, the text definition of building height is: the average maximum vertical height of a building or structure measured at a minimum of three equidistant points from mid-point of foundation to the highest point on the building or structure along each building elevation. Architectural elements that of a building or structure, such as parapet walls, chimneys, vents, and roof equipment are not considered part of the height of a building or structure.

2.3 3D object model

Considering the geometry, the city objects can be classified as Solid (Block) or Surface object (Zlatanova et al., 1996). The typical block objects are buildings (Zlatanova , Bandrova, 1998). Extruding footprints into 3D block-shaped models is one of the well-known, simple, and easy to implement methods to construct 3D city models (Ledoux , Meijers, 2009). Surface objects are all artificial and natural features which are integrated into the terrain, e.g., streets, paths, passways, etc. In this paper, we focus on the buildings as block objects only. Further, the building part above terrain is named as 'air' body, while the part under is regarded as 'underground' body (Zlatanova , Tempfli, 1998).

2.4 Terrain

Terrain is referred as an area of land or a type of land when considering its physical features⁴. In digital domain, the Earth's surface is commonly modelled by means of DTM (Brandli, 1996), such as TIN, which is an efficient alternative to the dense grid digital elevation model (DEM) to present terrain surface (Lee, 1991). It represents a surface formed of non-overlapping contiguous triangular facets that are with irregular sizes and shapes, which can in an intuitive way describe the continuous elevation changes of terrain. There are other alternative names, which also may represent different products, such as digital elevation models (DEMs), digital height models (DHMs), digital ground models (DGMs) as well as digital terrain elevation models (DTEMs).

¹<https://encyclopedia2.thefreedictionary.com/Building+footprint>

²<https://www.cob.org/documents/planning/applications-forms/building-height-calculation.pdf>

³http://centralpt.com/upload/375/4785_Buidling%20Height%20Definition.pdf

⁴<https://www.collinsdictionary.com/dictionary/english/terrain>

3. APPROACHES FOR INTEGRATING 3D OBJECTS AND TERRAIN

3.1 Footprints + DTM

As mentioned above this option is largely used as the footprints and DTMs are commonly available as governmental and open data sets (e.g., OSM). If the data are obtained from different sources, they need to be transformed (aligned) in the same coordinate system. Then, the process consists of four steps as follows (Figure 4):

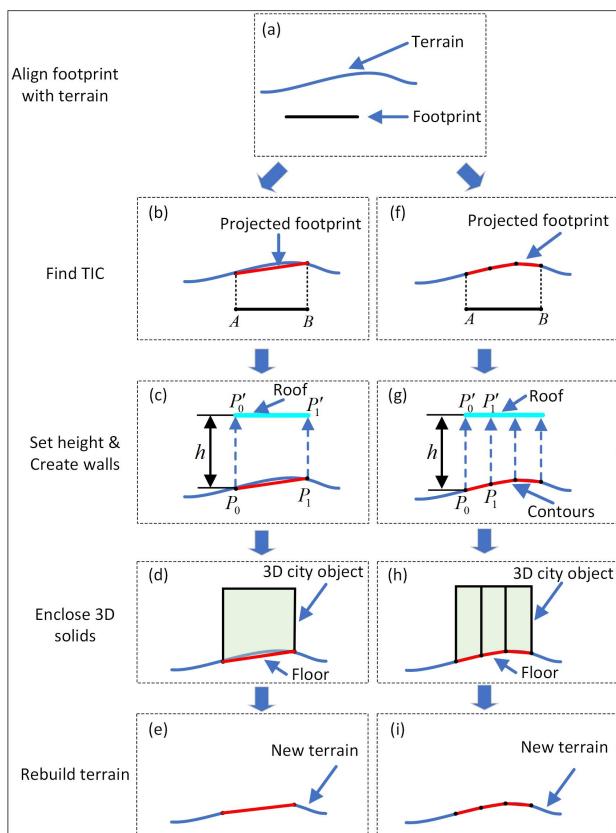


Figure 4. The process of 3D modelling based on footprint and DTM.

- Find TIC by projecting footprints onto the terrain;
- Set height and create walls' surfaces of the 3D buildings;
- Generate roof and ground surface to create 3D building (solids);
- Rebuild terrain considering TIC as constraints.

Step 1: Find TIC by projecting footprints onto the terrain

The footprint is normally presented as a 2D polyline (polygon), whose vertices are coplanar. Therefore, the first step is to find corresponding building footprints projection on the terrain. This corresponding building footprint is actually the needed TIC. Each composing polygon should follow counter-clockwise orientation to make sure the final 3D building objects are correct solids.

Two methods are presented in this step. The first method is Projecting Vertices of footprints on the Terrain (PVT); the

second is Projecting the whole footprint Polyline on the Terrain (PPT). The first method ensures the number of points of original footprints is preserved (Figure 4 (b)). The second method ensures the curvature of the terrain is considered (Figure 4(f)). The first approach results in less points and therefore less facets on the walls, whereas the terrain is changed. The second approach might lead to unnecessarily complex walls, which differ from the reality. However, the terrain is preserved, which might be important for some applications. As expected, the projected footprints shape is unaltered as indicated from the top view.

Step 2: Set height and create walls' surfaces of the 3D buildings

After getting each counter-clockwise orientated projected footprint, walls of this building are created by extruding them up along the $Z - direction$ based on h (Figure 4(c) and (g)). In particular, a line segment of a projected footprint, which has the start and end vertices $P_0(x_0, y_0, z_0)$ and $P_1(x_1, y_1, z_1)$, respectively. P_0 and P_1 will be extruded up along the $Z - direction$ based on h to get $P'_0(x_0, y_0, h)$ and $P'_1(x_1, y_1, h)$. Then, these four vertices can form a wall, which can be represented as a $polyline(P_0, P_1, P'_1, P'_0, P_0)$.

As mentioned previously, the two different projection approaches would result in two different models, for the PVT case, one polygon represent a wall, while for the PPT case, several polygons are combined together to form a wall (Figure 4(d) and (h)).

Step 3: Generate roof and ground surfaces to create 3D buildings (solids)

The third step is to enclose generated walls as 3D buildings (solids) by adding roof and floor. All roofs are considered as planar surfaces, and they are generated by making a polygon surface from a set of vertices. The floor is slightly different from roof, because vertices used for the floor are not coplanar, so connecting the projected vertices can get an enclosed 3D polyline, and then patching it as a surface. It should be noted that all vertices from one floor should be ordered in clockwise orientation before constructing the floor. After getting the roof and floor, each 3D building can be enclosed by joining surfaces.

Step 4: Rebuild terrain considering TIC as a constraint

To keep the consistency between 3D buildings and terrain, the DTM has to be re-computed considering the vertices and edges of TIC (projected building footprints) as constraints (Figure 4(e) and (f)). In particular, the re-computation process takes the vertices of projected footprints as *Points*, and their edges as *Breaklines* (constraints) to calculate a Constrained Delaunay Triangulation (CDT) to represent the terrain.

3.2 Footprints + Point Cloud

With the wide application of laser scanning technology, 3D point clouds become a data source for 3D city objects (Hammoudi et al., 2009). Hence, footprints and point clouds are also a possible input combination to construct 3D models and terrain. The proposed approach includes four steps (Figure 5):

- Classify point clouds with the help of footprints;
- Compute roof outline;

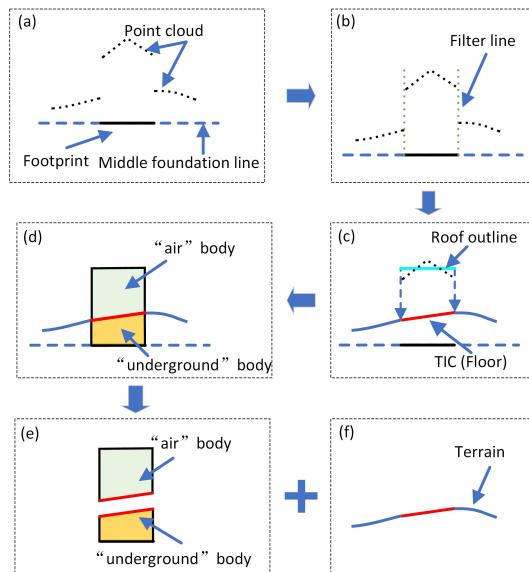


Figure 5. The process of modelling based on footprint and point cloud.

- Project roof outline onto terrain to get TIC and 3D building (solid);
- Rebuild terrain.

Step 1: Classify point cloud based on footprints

This approach assumes the original point clouds are unclassified. Footprints are afterwards used to group point clouds for each building. The terrain can be also derived from the point clouds (Zhang et al., 2003).

Step 2: Compute roof outline

After grouping the point clouds, the average Z of roofs will be calculated. Then, using the Z of roof points and the footprint, we can get the outline of a roof. It should be mentioned that this step actually simplifies the roof as a horizontal planar, rather than pitched. Surely, if the roofs are represented as tilted polygons matching the point cloud distribution, the final results of 3D buildings will be more accurate.

Step 3: Project roof outline onto terrain to get TIC and 3D building (solid)

This step is projecting the roof outline onto the terrain along the $Z - direction$ to get the TIC. We can get the wall surface of "air" body by connecting the planar roof with floor contour (generated based on TIC) (Figure 5(c)). The floor contour will be further patched into floor surface, as is the roof. Combining the roof surface, side walls, and floor surface, the "air" body of the building are created. If we have point clouds demarcating the underground, they can be processed with the similar steps to get the "underground" body (Figure 5(d) and (e)).

Step 4: Rebuild terrain

The last step is using the TIC as *Breaklines*, and the terrain point clouds as *Points* to rebuild a CDT (Figure 5(f)).

3.3 3D building model + DTM

This method of creating 3D building model is very popular in current applications for the creation of large-scale 3D models, because it is one of simplest way to generate them as extruded footprints to a certain height (Ledoux , Meijers, 2009). Also, the BIM/CityGML are two popular sources of 3D building models, which has been widely used in different domains, such as urban planning, city analytics (Biljecki et al., 2015, Afroz et al., 2018).

This approach taking 3D building model and DTM as inputs includes four steps (Figure 6).

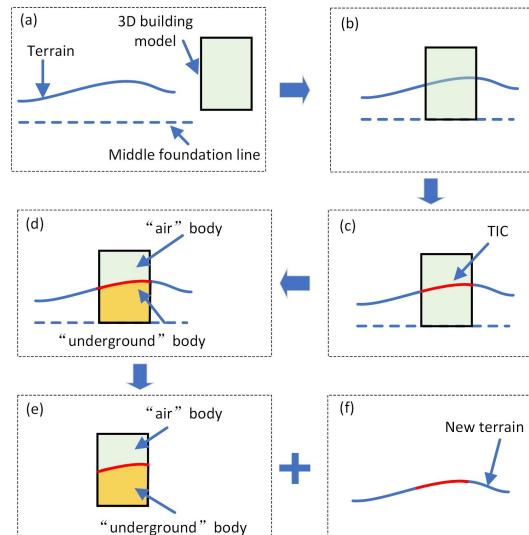


Figure 6. The process of modelling based on 3D building models and DTM.

- Align 3D building model with DTM;
- Get TIC by 3D intersection;
- Obtain "air" body, and "underground" body;
- Re-compute terrain.

Step 1: Align 3D building model with DTM

This alignment step is used to bring georeferencing information to 3D building models. In other words, if the 3D building models are georeferenced already or have georeferencing information (e.g., BIM models have a translation vectors), this step is not needed. Otherwise, this step is used to make sure that the 3D building models are at the right location, not only considering horizontal direction but also the vertical (Figure 6(b)).

Step 2: Get TIC by 3D intersection

After the alignment, the 3D building model will have an intersection with terrain geometrically. Thus, the TIC can be computed by 3D intersection between the 3D building models and terrain (Figure 6(c)).

Step 3: Obtain "air" body and "underground" body

The 3D building model are cut into "air" body and "underground" body (if it has) for some applications. For

instance, doing area budget for wall decoration, where the walls of “air” body are needed. Then, the TIC will be used to make a surface, and this surface will be used twice to cap the upper body and lower body as enclosed volumes. In other words, the “air” body and “underground” body are volumes, and the floor surface of the former and the roof surface of the latter is actually the surface patched from TIC (Figure 6(d) and (e)).

Step 4: Re-compute terrain

A new terrain will be computed based on the original DEM and TIC (Figure 6(f)). The geometry of DEM will be coerced into points or multi-points. Then, a CDT will be generated by taking the points beyond the TIC as *Points*, and TIC as *Breaklines*. This re-computation keeps original shapes of terrain except the part occupied by 3D objects.

3.4 3D building model + Point Cloud

When taking 3D building models and point clouds as the inputs, we have to keep in mind that at least one of them should be georeferenced and point clouds should be classified into different categories, such as building, terrain, and vegetation. Thus, footprints are needed to ensure that the 3D building models appear in the right locations, and point cloud representing roofs should be classified by footprints, which is similar to the Figure 5(b).

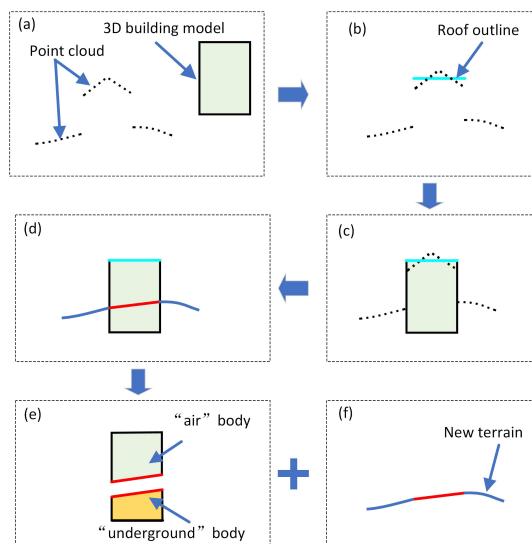


Figure 7. The process of modelling based on 3D building models and point clouds.

With georeferenced 3D building models, and classified point clouds, the proposed approach includes four steps (Figure 7):

- Estimate outline of a roof face;
- Align 3D building models with roof outlines;
- Construct initial terrain and TIC;
- Rebuild 3D building models and terrain.

Step 1: Estimate outline of a roof face

With the classified roof point clouds, it is possible to estimate the orientation and height of a roof face accurately, but the

outline of a roof face is a difficult issue (Vosselman , Dijkman, 2001). Therefore, we simplify this step by representing all roofs as planar surfaces. Thus, for each roof, its point cloud will be used to compute a planar roof based on the average Z (Figure 7(b)).

Step 2: Align 3D building models with roof outlines

Then, 3D building models will be aligned based on their corresponding roof lines. If the roof of a 3D building is planar, it will be moved to overlap with roof lines. Otherwise, for a pitched roof, the middle line of the roof in the Z – direction will be used to align with the roof line.

Step 3: Construct initial terrain and TIC

For the terrain point cloud part, an initial terrain can be constructed by using triangulation for instance. With these alignments, the 3D building model will have an intersection with the terrain. Thus, the TIC can be computed by 3D intersection between the 3D building model and terrain (Figure 7(d)).

Step 4: Rebuild 3D building models and terrain

The last step is recomputing a 3D building model as ‘air’ body and ‘underground’ body (if it has) based on TIC (Figure 7(e)). Moreover, a CDT will be re-computed as new terrain by taking the terrain point cloud as *Points*, and TIC as *Breaklines*(Figure 7(f)).

3.5 Point Cloud

The consideration of taking only point clouds as data source for both 3D buildings and terrain assumes that point clouds are georeferenced and classified into at least two categories, buildings and terrain. If not, point clouds should be georeferenced and classified beforehand.

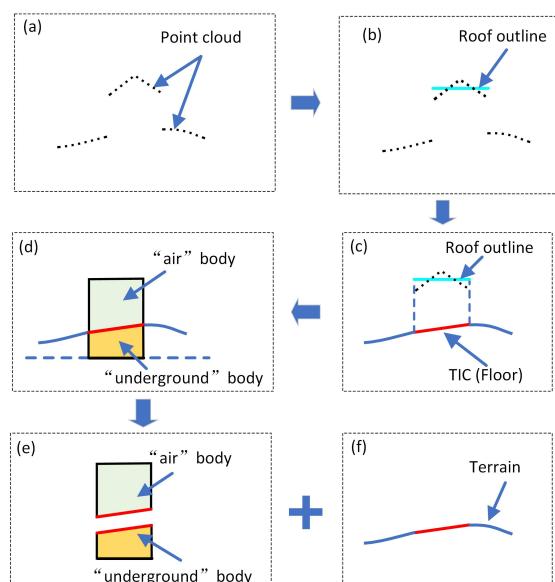


Figure 8. The process of modelling based on point cloud only.

With the georeferenced and classified point cloud, the whole process to construct 3D building models, TIC, and terrain requires four steps (Figure 8):

- Obtain outline of a roof face;
 - Compute original terrain and TIC;
 - Re-compute 3D buildings (solids);
 - Rebuild terrain.

Step 1: Obtain outline of a roof face

In this approach, we also simplify all roofs as planar surfaces. Thus this step is the same as the first step of the previous approach (*3D building model + Point Cloud*) (Figure 8(b)).

Step 2: Compute original terrain and TIC

An original terrain can be computed by using terrain point cloud. Then, we can project roof outlines onto terrain along the Z -direction to get the TIC (Figure 8(c)).

Step 3: Re-compute 3D buildings (solids)

Then, walls of "air" body can be obtained based on the roof lines and TIC. The TIC will be further patched into floor surface and roof surface for the "air" body and "underground" body respectively (Figure 8(d) and (e)).

Step 4: Rebuild terrain

The last step is also recomputing a CDT as new terrain by taking the terrain point cloud as *Points*, and TIC as *Breaklines* (Figure 8(f)).

4. DISCUSSION OF DIFFERENT APPROACHES

The approach footprint + DTM is the easiest one to be implemented. The only critical information is the height of buildings for this approach, and it can be determined from different data, such as point cloud, OSM, or mapping data. Moreover, comparing the two projection methods, we can conclude that the details of building footprints and terrain are mutually influenced. To be more specific, detailed footprints can lead to more complex terrain and vice versa. Keeping detailed footprints or terrain depends on the applications. For instance, the building management task of estimating the cost for cleaning windows for a high-rise building would require calculations undertaken using the “air” body and it is better to keep detailed building footprints. Another example is using the re-computed terrain for navigation and wayfinding purposes, where keeping the detailed terrain is a better choice.

All approaches using the point cloud as an input sharing the common challenge in how to accurately classify point clouds into building, terrain, and other city objects. Based on the proper classification of point clouds, another difficulty is to estimate/extract accurate roof outlines (especially the pitched roofs) for 3D building reconstruction. In other words, the approaches using the point cloud as input in this research assume that the processing of the point clouds itself is already very accurate. In our case, the research represents a specific application of point cloud. The disadvantage of using point clouds could be inaccurate preprocessing of point clouds which would bring afterwards additional problems related to the accuracy of the proposed methods.

The DTM constructed from point clouds is more accurate than that from the DEM, because the latter has approximated the terrain to some extent.

Re-computing the 3D building models as an ‘air’ body and an ‘underground’ body depends on the 3D object models themselves and applications. For instance, a BIM model definitely cannot be recomputed into two parts based on TIC, while block-shaped models extruded from footprints can.

The *Points* for terrain re-computation should be considered based on the specific applications in questions. The simplest terrain is a CDT that only takes the vertices of TIC as *Points* and TIC as *Breaklines*. This terrain can be used for a large area precinct visualization, but it is not good for some fine scale applications, for instance, the disability access and navigation tasks involving for example people using wheelchairs. In such use cases changes in slopes across the terrain are critical and thus accurate terrain between the 3D buildings is more important.

5. EXPERIMENTS AND DEMONSTRATIONS

To demonstrate and evaluate the entire process of the proposed approaches, we implement the first approach (footprints + DEM) by taking a real world precinct example, the University of New South Wales (UNSW) Kensington campus in Sydney. There are 98 building footprints (Figure 9), where footprints are coloured in red. Elevation-wise, the campus can be distinguished as lower and upper campus. The building footprints are provided by the Estate Management of the UNSW as a CAD file without any attributes attached to the geometry. To add attributes, we used the function *Join attributes by location* implemented in QGIS to copy the attributes (such as *_uid*, *id*, *osm_id*, *osm_way_id*, *name*, etc.) of footprints from OSM. Also, we add the attribute *height*, which is used as the height of the buildings.

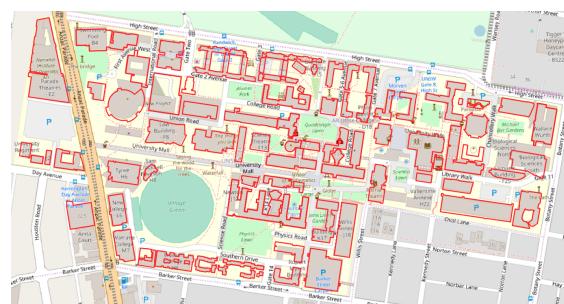


Figure 9. The 2D map and building footprints of the UNSW campus.

Three software packages are used in the experiment: QGIS, Rhinoceros (with Grasshopper), and the Feature Manipulation Engine (FME). QGIS is used to add attributes, set the coordinate system, and edit the building footprints. Rhinoceros (with Grasshopper) is used to process the entire workflow except the terrain re-computation. FME is used for extract point clouds from the DEM, and to conduct terrain re-computation, and visualization.

A DEM (5 metre) of experimental area is obtained from Elevation Foundation Spatial Data⁵ sourced from the National

⁵<http://elevation.fsdf.org.au/>

Elevation Data Framework (NEDF), Geosciences Australia. The geometry of the DEM is coerced into point clouds, and then, they are used to construct Delaunay triangulation representing the terrain (DTM) (Figure 10).

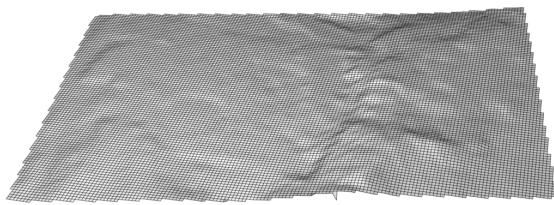


Figure 10. The terrain of the UNSW campus.

After aligning the building footprints with terrain based on their geolocation, the building footprints are projected onto the terrain. As mentioned above, there are two methods to get the projected footprints (Figure 11), PVT and PPT. From the top view, the shape of projected footprints is the same as the original, but the results of PVT are 3D polylines, while the PPT method are mesh surfaces.

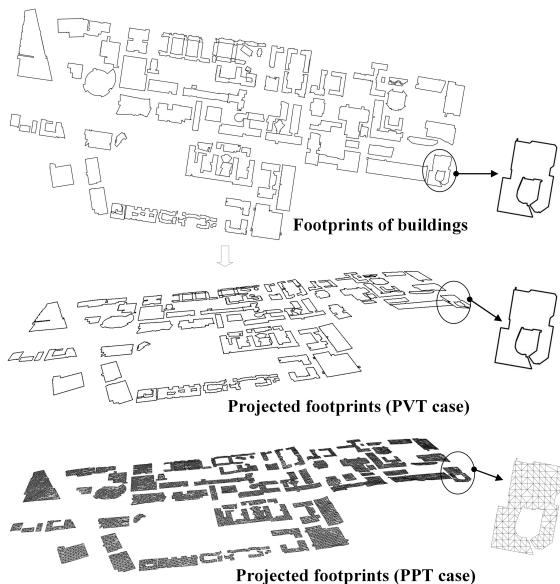
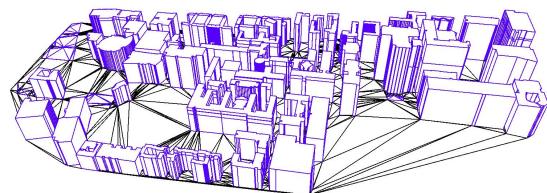


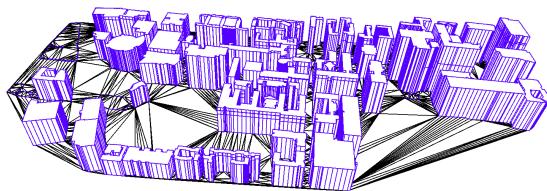
Figure 11. The projected footprints on the terrain.

Before the walls generation procedure, 3D polylinies (TIC) of the projected footprints are extracted. For the PVT method, the projected footprints are already captured as 3D polylinies, but for PPT method, the edges with a single adjacent face will be extracted. Then, each line segment of a polyline will be extruded up to the same height along the $Z - direction$ based on building heights to get a quadrilateral side walls. The floor is made by creating a fragmented patch surface by taking the 3D polyline as the boundary. The component named *Fragment Patch* in Grasshopper is used, which takes *Fragment polyline boundary (curve)* as inputs, and *Fragmented patch* as outputs. The roof is a planar surface constructed by ordered 3D points.

The final step is to re-compute terrain using the vertices and edges of the projected building footprints as constraints. We used the function of the transformer (*TINGenerator*) from FME, where the vertices of projected footprints are used as *Points/Lines*, and their edges are *Breaklines*.



(a) PVT method



(b) PPT method

Figure 12. The 3D buildings with terrain.

The two projection methods bring two different results related to the 3D building objects, and the terrain (Figure 12). Comparing terrain of the two methods, the result of PVT has less vertices and triangulations in the terrain, which means that the changes of terrain are more than in the PPT method. The results of the re-triangulated DTM show that footprints are clearly recognisable within the triangulated surface. The PVT method is more efficient in one aspect, having less vertices and edges, which is beneficial for a large precinct scale area. In contrast, the terrain of PPT methods has more vertices and edges, simplifying the terrain less, which is good for more precise navigations. Both methods have a similar principle in changing the 3D building objects and terrain slightly to fit each other, thus, the topological issue can be fixed.

6. CONCLUSION AND FUTURE WORK

As we embark upon the Digital Twin representation of our precincts and cities to support planning, analytics and community engagement, there is a need to be able to generate fit for purpose 3D models which suitably integrate 3D objects with terrain. In this paper, we have discussed topological issues that can occur when integrating 3D objects with terrain. The key to solving this issue lies in obtaining correct TIC and amending 3D objects and the terrain properly based on it. There are five combination method to construct 3D objects and terrain, where all cases can be successfully resolved if a correct procedure is followed. We have implemented the most commonly used method (Footprint + DTM) to illustrate the procedure. This method provides the most simplified version of terrain. Some parts of the terrain without having enough points, in between footprints, can be poorly presented. In such cases, the terrain can be improved by adding more terrain points either from the original terrain or integrating more footprints (e.g. streets, trees, lamp post).

In future work, we will consider more cases of TIC computation in which high-accuracy terrain is integrated with solid or surface objects and the accuracy of the terrain has to be preserved. Specific attention will be given on integration of BIM models and terrain. BIM models may require very complex TIC and significant modification of the surrounding terrain. BIM models are commonly not georeferenced, which

means that the TIC can also vary with respect to the altitude that the BIM model has to be located on the terrain.

ACKNOWLEDGEMENTS

This research (No.RP2011u1) is funded by the CRC for Low Carbon Living Ltd supported by the Cooperative Research Centres program, an Australian Government initiative. The publication acknowledges the contribution of all authors, researchers, participants and others as appropriate. This work is supported by the program of China Scholarships Council (No.201606410054).

REFERENCES

- Afrooz, Aida E, Lowe, Russell, Leao, Simone Zarpelon, Pettit, Chris, 2018. 3d and virtual reality for supporting redevelopment assessment. *Real Estate and GIS*, Routledge, 162–185.
- Batty, Michael, 2018. Digital twins. *Environment and Planning B: Urban Analytics and City Science*, 45, 817–820.
- Beil, Christof, Kolbe, Thomas H, 2017. Citygml and the streets of new york-a proposal for detailed street space modelling:(accepted). *Proceedings of the 12th International 3D GeoInfo Conference 2017*, 9–16.
- Biljecki, Filip, Stoter, Jantien, Ledoux, Hugo, Zlatanova, Sisi, Çöltekin, Arzu, 2015. Applications of 3D city models: State of the art review. *ISPRS International Journal of Geo-Information*, 4, 2842–2889.
- Brandli, MARTIN, 1996. Hierarchical Models for the Definition and Extraction of Terrain. *Geographic objects with indeterminate boundaries*, 2, 257.
- Brédif, Mathieu, Tournaire, Olivier, Vallet, Bruno, Champion, Nicolas, 2013. Extracting polygonal building footprints from digital surface models: A fully-automatic global optimization framework. *ISPRS journal of photogrammetry and remote sensing*, 77, 57–65.
- Fan, Hongchao, Zipf, Alexander, Fu, Qing, Neis, Pascal, 2014. Quality assessment for building footprints data on OpenStreetMap. *International Journal of Geographical Information Science*, 28, 700–719.
- Fruh, Christian, Zakhori, Avideh, 2001. 3D model generation for cities using aerial photographs and ground level laser scans. *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, 2, IEEE, II–II.
- Gröger, Gerhard, Kolbe, Thomas H, Czerwinski, Angela, Nagel, Claus, 2008. OpenGIS city geography markup language (CityGML) encoding standard, version 1.0. 0.
- Haala, Norbert, Brenner, Claus, 1997. Generation of 3d city models from airborne laser scanning data. *Proceedings EARSEL Workshop on LIDAR remote sensing on land and sea, Tallinn/Estonia*.
- Hammoudi, Karim, Dornaika, Fadi, Paparoditis, Nicolas, 2009. Extracting building footprints from 3D point clouds using terrestrial laser scanning at street level. *ISPRS/CMRT09*, 38, 65–70.
- Ledoux, Hugo, Meijers, Martijn, 2009. Extruding building footprints to create topologically consistent 3d city models. *Urban and Regional Data Management*, CRC Press, 51–60.
- Lee, Jay, 1991. Comparison of existing methods for building triangular irregular network, models of terrain from grid digital elevation models. *International Journal of Geographical Information System*, 5, 267–285.
- Li, Zhilin, Zhu, Qing, Gold, Chris, 2004. *Digital terrain modeling: principles and methodology*. CRC press.
- Pettit, C, Widjaja, I, Russo, P, Sinnott, R, Stimson, R, Tomko, M, 2012. Visualisation support for exploring urban space and place. *International Society for Photogrammetry and Remote Sensing*.
- Shiode, Narushige, 2000. 3D urban models: Recent developments in the digital modelling of urban environments in three-dimensions. *GeoJournal*, 52, 263–269.
- Trubka, Roman, Glackin, Stephen, Lade, Oliver, Pettit, Chris, 2016. A web-based 3D visualisation and assessment system for urban precinct scenario modelling. *ISPRS Journal of Photogrammetry and Remote Sensing*, 117, 175–186.
- Verma, Vivek, Kumar, Rakesh, Hsu, Stephen, 2006. 3d building detection and modeling from aerial lidar data. *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, 2, IEEE, 2213–2220.
- Vosselman, George, Dijkman, Sander, 2001. 3D building model reconstruction from point clouds and ground plans. *International archives of photogrammetry remote sensing and spatial information sciences*, 34, 37–44.
- Xie, Feifei, Lin, Zongjian, Gui, Dezhu, Lin, Hua, 2012. Study on construction of 3D building based on UAV images. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 39, B1.
- Zhang, Keqi, Chen, Shu-Ching, Whitman, Dean, Shyu, Mei-Ling, Yan, Jianhua, Zhang, Chengcui, 2003. A progressive morphological filter for removing nonground measurements from airborne LIDAR data. *IEEE transactions on geoscience and remote sensing*, 41, 872–882.
- Zhang, Keqi, Yan, Jianhua, Chen, S-C, 2006. Automatic construction of building footprints from airborne LIDAR data. *IEEE Transactions on Geoscience and Remote Sensing*, 44, 2523–2533.
- Zlatanova, Sisi, Bandrova, Temenoujka, 1998. User requirements for the third dimensionality. *E-mail Seminar of Cartography*, 1, 61–72.
- Zlatanova, Sisi, Tempfli, Klaus, 1998. Data structuring and visualization of 3d urban data: some aspects of the doctoral research: 3d gis for urban development. *Proceedings Agile conference, Enschede, The Netherlands 1998*, AGILE.
- Zlatanova, Siyka, 2000. Toward a 3D GIS for local governing: a web-oriented approach. *Proceedings of UDMS 2000, 22nd Urban Data Management Symposium, Delft, September 11–15, 2000*, Urban Data Management Society, VI33–VI40.
- Zlatanova, Siyka, Gruber, Michael, Kofler, M, 1996. Merging DTM and CAD data for 3D Modeling purposes for Urban Areas. *International Archives of Photogrammetry and Remote Sensing*, XXXI, 311–15.