

## DEFINITION OF A TRANSITION SURFACE WITH THE PURPOSE OF INTEGRATION BETWEEN A LASER SCANNER 3D MODEL AND A LOW RESOLUTION DTM

G. Agugiaro<sup>a,\*</sup>, T.H. Kolbe<sup>b</sup>

<sup>a</sup> Università di Padova, Laboratorio di Rilevamento e Geomatica, Via Marzolo 9, 35121 Padova, Italy - [giorgio.agugiaro@unipd.it](mailto:giorgio.agugiaro@unipd.it)

<sup>b</sup> TU Berlin, Institut für Geodäsie und Geoinformationstechnik, Straße des 17. Juni 135, 10623 Berlin, Germany - [kolbe@igg.tu-berlin.de](mailto:kolbe@igg.tu-berlin.de)

Commission V, WG V/4

**KEY WORDS:** DTM, laser scanner, integration, surface modelling, cultural heritage, triangulation, simplification

### ABSTRACT:

Thanks to quickly spreading technologies like laser scanning, which are becoming a quite common means of data acquisition for architectural objects or cultural heritage sites (but not only!), integration between datasets of different origin and resolution is still an open problem. This paper describes an approach whose goal is to define a surface which models a proper transition between a high resolution, laser-scanner-acquired model and a low resolution digital terrain model (DTM), by means of some “extra” information around the high resolution object as sort of “collar”. This information is generally present in laser scanner models and instead of pruning it during point cloud editing, we use it for our modelling purposes.

We present a (so far) deterministic approach, some initial results and discuss still unresolved issues and future improvements.

### KURZFASSUNG:

Dank der schnellen Verbreitung von Technologien wie z.B. Laser-Scannen, die zur Vermessung architektonischer Objekte oder Kulturgüter immer häufiger Anwendung finden, bleibt die Integration von Datensätzen unterschiedlichen Ursprungs und verschiedener Auflösung ein noch ungelöstes Problem. In diesem Artikel wird ein Ansatz beschrieben, dessen Ziel die Definition einer Übergangsfläche zwischen einem hoch aufgelösten, durch Laser-Scanner aufgenommenen Model und einem gröberen digitalen Geländemodell (DGM) ist, indem einige zusätzliche Informationen um das hoch aufgelöste Objekt so wie ein „Kragen“ verwendet werden. Diese weiteren Daten sind im Allgemeinen in den Laser-Scanner-Datensätzen bereits vorhanden und werden hiermit zum Zweck der Oberflächenmodellierung genutzt, statt – wie sonst üblich – in der Editierungsphase eliminiert zu werden.

Ein deterministischer Ansatz wird, zusammen mit den ersten Test-Ergebnissen, vorgestellt. Offene Probleme und künftige Verbesserungen werden angesprochen.

### 1. INTRODUCTION

Use of high resolution sensors like laser scanners is becoming more and more common for surveying tasks, nearly every single application field is in fact gaining benefits from these booming methodologies.

Moreover, the continuously increasing availability of multiple models of the same “object” with different resolutions or coming from different sensors leads to an array of still unsolved problems concerning their integration in terms of precision, data structure or simply geographical extension (just to name some). Projects like Google Earth are starting to provide world-wide coverage with multi-scale and multi-format geodata, but are still facing problems regarding, for example, embedding of high resolution objects (landmarks, city models, etc.) in a generally low resolution Digital Terrain Model (DTM).

This is, nevertheless, an open problem which can be found at any scale: we can think of high resolution models which are being produced nowadays in the framework of architectural objects or cultural heritage sites. Some particularly elaborated architectural details might have been acquired singularly and

need to be properly merged into the general model, thus facing possible geometric, topologic or semantic incompatibilities.

According to Laurini (1998) errors resulting from merging different datasets can be grouped into two groups: “layer fragmentation” errors originate in case of datasets covering the same region but containing different feature classes (e.g. a DTM and a “flying” 3D-building); “zonal fragmentation” errors refer instead to datasets containing the same feature class but covering spatially disjoint regions (e.g. overlaps or gaps at the borders of two adjacent DTMs).

In this paper we focus on some geometric and topological issues concerning zonal fragmentation and present a new approach of data integration between a laser-scanner-acquired, high resolution model and a lower resolution DTM. We further present some initial test results coming from real datasets.

Our goal is to create a topologically correct and geometrically “clean” transition surface between the two models, which is influenced by local parameters from both high and low resolution datasets. This surface allows for a smooth transition between them, not only in geometric terms but also precision and data density.

\* Corresponding author

## 1.1 Related work

Research work on data integration tailored to DTM is a hot topic and embraces several heterogeneous approaches. Since DTM enhancements or update operations might apply to some portions of a dataset only, Karel & Kraus (2006) deal, for instance, specifically with DTM quality assessment and present diverse local measures for the quantification of DTM quality. In particular, the distribution of the data (with regard to density, completeness and type), accuracy of the measurements and consistency of the data are investigated and discussed.

Koch (2005) has proposed an algorithm which uses semantic information in order to enhance DTMs using data coming from a two-dimensional topographic vector dataset, since some geographical features like roads, rivers, and lakes contain intrinsic height information which might be employed through a proper set of constraints. A lake represents for instance a horizontal plane, therefore the neighbouring banks must be higher than the water; in the same way, the slope of a road must not exceed a certain maximum value.

Other authors (Schmittwilken *et al.*, 2007) attempt to capture semi-automatically detailed building models aiming particularly at the building “collars”, i.e. the transition from facades to the digital elevation model, using hybrid data sources.

Stadler and Kolbe (2007) discuss and analyse different data qualities with respect to their semantic and spatial structure. Semantic information, which has been coherently structured with respect to geometry, can help to reduce the ambiguities for geometric integration.

Multiple DTMs can already be stored and represented in CityGML (Gröger, Kolbe *at al.*, 2008), an open data model and XML-based format for the exchange of 3D city models. They can be restricted to specific regions by means of a validity extent polygon, which can have holes (thus allowing nested DTMs), but a method for their integration has not been presented yet.

## 2. TRANSITION SURFACE

### 2.1 Initial assumptions and properties

We assume both datasets to be triangulated meshes, both georeferenced and previously aligned, and concentrate our attention on the overlapping zones between the DTM and the high resolution model. We further assume the high resolution model to have been acquired with some extra information surrounding the actual surveyed object (which is common, for example, when working with laser scanners). Thus, the extents of the dataset are actually larger than strictly needed and, like mentioned before, we can think of a sort of “collar” around the scanned object. Generally, this extra information is pruned during the following point cloud editing; we use it instead for our modelling purposes and leave the real high resolution object actually unchanged. We finally assume that (at least) the collar is 2.5D, i.e. for each planimetric coordinate, only one height value is given.

Let’s take as an example the two models shown in Figure 1: we have a low resolution DTM of an archaeological excavation field and a high resolution triangulated model, which has been georeferenced and aligned to the DTM. The high resolution model represents a pit that was successively dug and is therefore more recent in time. The two models define three zones that can be described as follows:

- The “real” high resolution object (the pit), above which old DTM data refers to portion of the field that do not exist anymore. In the final integrated model, these points will be absent. The high resolution object in this zone can be of any type (2.5D or 3D, e.g. including overhangs), since it won’t be actually modified.
- The “far away” DTM points, which will remain in the final model
- The overlapping zones, which contains data from both the high and low resolution models. By hypothesis, the two models have been previously georeferenced and aligned, so points in this zone refer to the same object (it is the “collar”). However, we can encounter slight differences, for example with regard to the height profile.

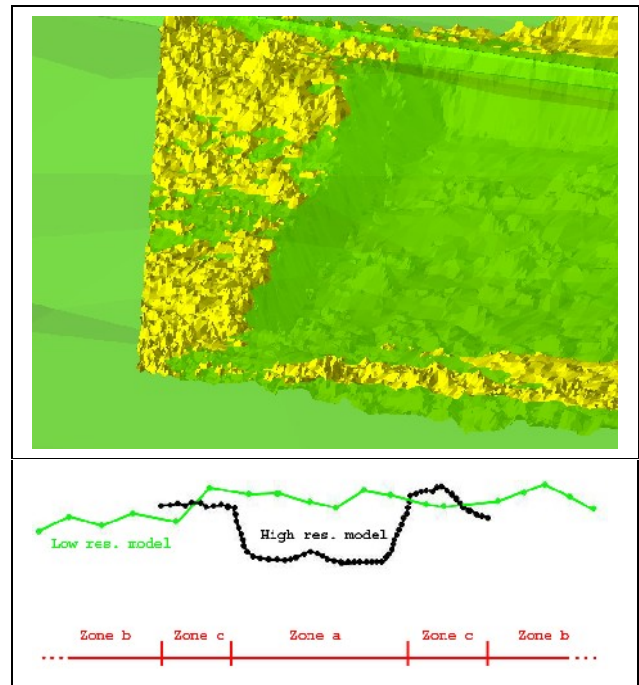


Figure 1. Zonal fragmentation: a low resolution DTM (green) and a high resolution, laser-scanner-acquired model (yellow/black). Three different zones result from these two overlapping models.

We will focus our attention on this last zone c), the overlap area, thus leaving the high resolution object and the “far away” DTM unchanged.

In the overlapping zone we want to define a transition surface, which connects geometrically and topologically both models and which must have following properties. The transition surface:

- is limited inwards by an inner border, which is connected with the high resolution object
- is limited outwards by an outer border, which is connected with the low resolution DTM
- may contain both high and low resolution data
- does not modify any height value outside its borders
- must guarantee at least C0 (height) and C1 (tangential) continuity at the borders
- guarantees a smooth transition also in terms of point density. This can be achieved through a progressive simplification of the transition surface mesh. We can thus expect triangles of growing size while we move along the transition surface from the inner border to the outer border (Figure 2).

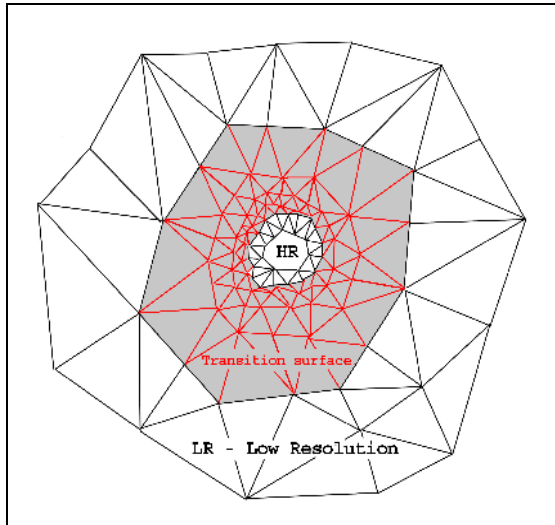


Figure 2. The transition surface is intended to guarantee a smooth transition between the high resolution and the low resolution models also in terms of point density.

This transition surface is obtained in the following steps: both datasets are first characterised by means of local parameters, the extents of the overlapping area (i.e. the borders) are defined and a new triangulation inside the so defined domain is carried out. Secondly both a height interpolation model and the accompanying distance-dependent weights are formulated, which determine the new height profile in the transition zone. Finally, the mesh is progressively simplified with respect to a function which influences the decreasing density between high and low resolution datasets.

## 2.2 Extents

Characterisation of the input datasets is carried out by means of local parameters, whose goal is to describe the distribution of quantities like point density and average distance from neighbouring points. For every dataset, these local parameters are calculated on the triangulated surface itself (and not on the xy-plane). Furthermore, in order to give a quality statement within the common overlapping areas, local height differences between the two models ( $\Delta z$ ) are computed on the z-axis on a per point basis (Figure 3).

The height value of every projected point is obtained through either bilinear interpolation on the triangle surface or linear interpolation in case of intersection with an edge.

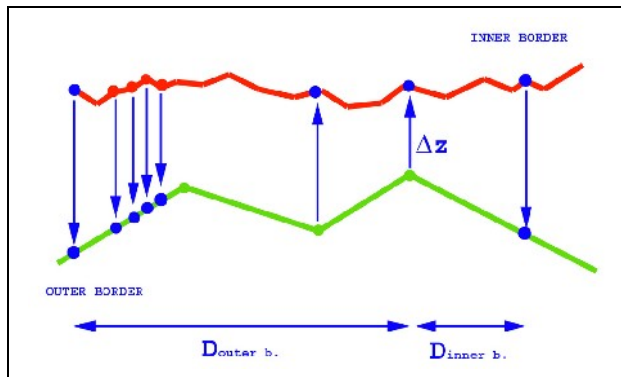


Figure 3. Height differences ( $\Delta z$ ) between the two models are obtained projecting high resolution points on the low resolution mesh and vice versa.

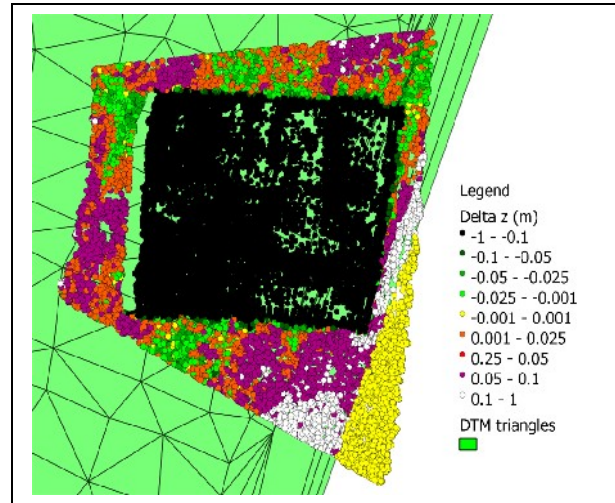


Figure 4. Map representing classified height differences ( $\Delta z$ ) between high resolution points and the low resolution mesh.

In case the borders of the overlapping zone are not given as an input, they can be selected (presently not yet automatically) by means of the  $\Delta z$  map like shown in Figure 4. In particular, the inner border is a closed 3D-polyline obtained from adjacent triangle edges of the high resolution mesh. This polyline separates the high resolution object from the collar and is intended to remain unchanged. The outer border is a closed 3D-polyline which originates from the high resolution mesh and is then projected (on the z-axis) on the low resolution mesh (Figure 5).

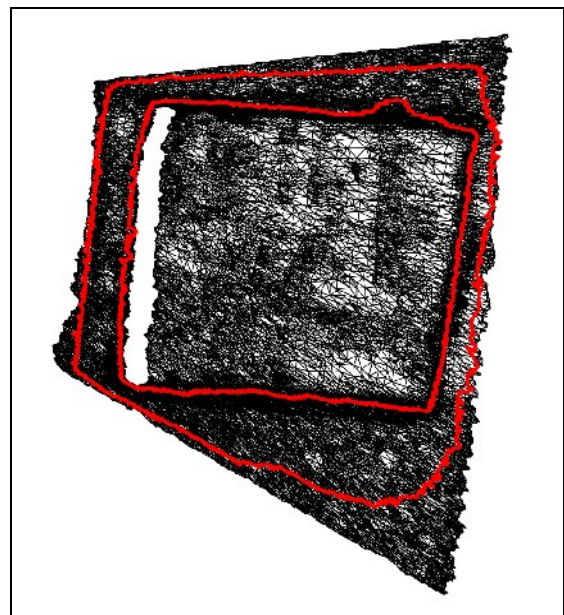


Figure 5. High resolution mesh with inner and (not yet projected) outer border.

The two borders define a domain in the xy-plane which may be of any form (not necessarily strictly convex) and which may contain one or multiple holes – depending on how many high resolution object share the same collar.

Inside this domain we create a new constrained Delaunay mesh using both high and low resolution points. The inner and outer border are treated as hard-breaklines and the external points of overlapping low resolution triangles are also included, although the latter choice is optional. However, doing so we guarantee a topologically continuous surface, because the outer border points are now connected to the low resolution ones. No other constraints are considered at this stage. This can be seen in Figure 6: heights of the points in the transition zone are not smoothed.

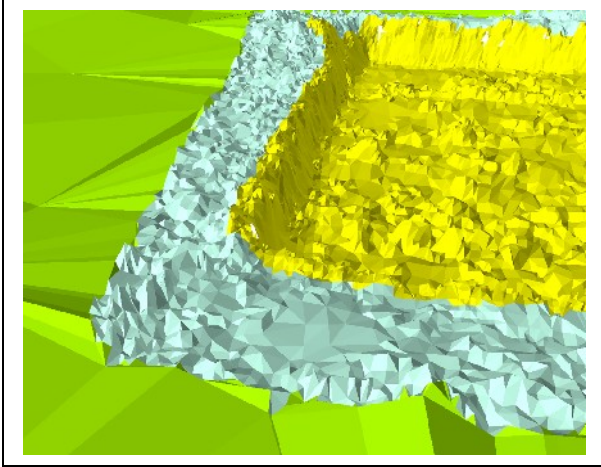


Figure 6. Constrained Delaunay triangulation inside the transition area (azure): height values have not been modified yet.

### 2.3 Distance-weighted height model

The next step for the creation of the transition surface is to define a proper height profile. Ideally, we expect the height values to be more “similar” to the DTM the closer we get to the outer border and – vice versa – to be more “similar” to the high resolution object, the closer we get to the inner border. Proper weights, which are distance-dependent and which will later be applied to the  $\Delta z$  values, are thus needed.

We define a  $[0,1] \times [0,1]$  domain, where the x-axis represents a normalised distance parameter and y-axis the weight values range. The height value of the low resolution model at the outer border corresponds here to the origin of the axis, and the height value of the high resolution model at the inner border corresponds to point (1,1).

Inside this square domain we can model different weight functions. The simplest, linear function  $y=f(x)=x$  allows for C0 continuity: points lying on the outer border ( $x=0$ ) have the same height as the DTM ( $y=0$ ), points on the inner border ( $x=1$ ) have the same height as the high resolution mesh. However this linear model doesn't guarantee C1 continuity at the border points (0,0) and (1,1), where the transition surface is joined with the models. An exponential function  $y=f(x)=x^a$  (with  $a>0$ ,  $a \neq 1$ ) allows for C0 and some C1 continuity, however the tangent value is acceptable only in either of the border points (depending on the value of parameter  $a$ ). A piecewise polynomial curve like a uniform spline of order 3 as

$$y = f(x) = \begin{cases} x^2 & 0 \leq x \leq 0.5 \\ -2x^2 + 4x - 1 & 0.5 < x \leq 1 \end{cases} \quad (1)$$

can provide the needed level of continuity. Some functions are plotted in Figure 7.

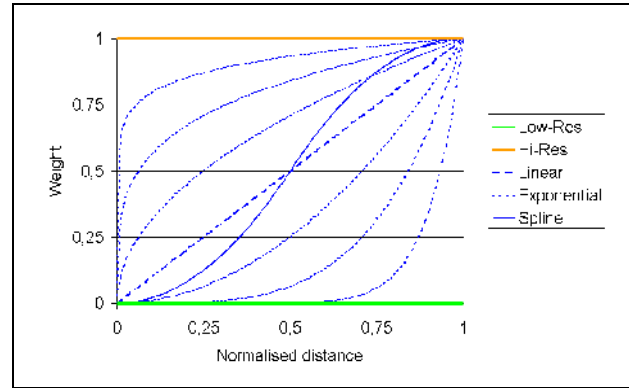


Figure 7. Different weight functions: linear, exponential (with varying exponential values) and spline.

Once the weight function has been chosen, a proper formulation of the  $x$  parameter has to be given. Since we must know for any point belonging to the transition surface its normalised distance between inner and outer border, we define

$$x = \frac{d_{outerBorder}}{d_{outerBorder} + d_{innerBorder}} \quad (2)$$

where:

$d_{innerBorder}$  = shortest distance between a point and the inner border with respect to the annular topology,

$d_{outerBorder}$  = shortest distance between a point and the outer border with respect to the annular topology.

In case of regular convex shapes with radial symmetry, computation of distance can be generally achieved with simple Euclidean distance functions; however this approach might lead to errors in case of extremely irregular shapes (concavities of the borders and/or presence of multiple holes, see the example shown in Figure 8). A general solution to this problem can be achieved if  $d_{innerBorder}$  and  $d_{outerBorder}$  are calculated separately through progressive iterative buffering, from the inner border outwards, and vice versa. A combination of the two resulting maps yields the total distance for any point inside the transition surface domain from the borders (Figure 9).

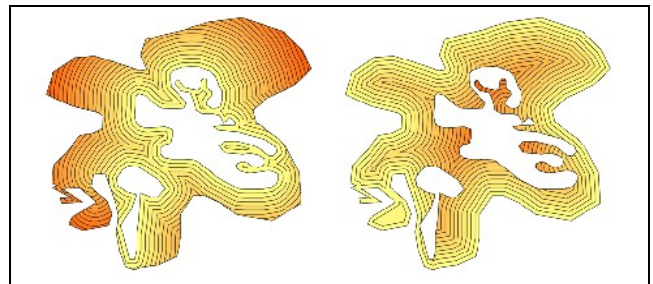


Figure 8. Example of progressive buffering from the inner borders outwards (left) and vice versa (right), inside an irregular shaped domain.

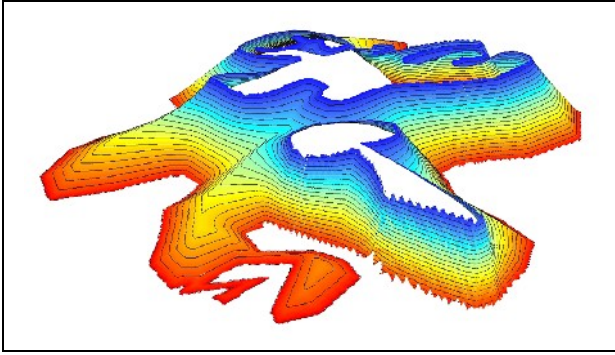


Figure 9. Three-dimensional representation of the combination of the two buffer maps in Figure 8, which yields the normalised distance for any point inside the irregular shape.

Once the  $x$  parameter and the weight function  $y=f(x)$  have been determined, the height profile inside the transition surface can be calculated: the new height value of any point will be  $z_{new}=\Delta z \cdot f(x)$ . Thus C0 and C1 continuity at the borders have now been added (Figure 10).

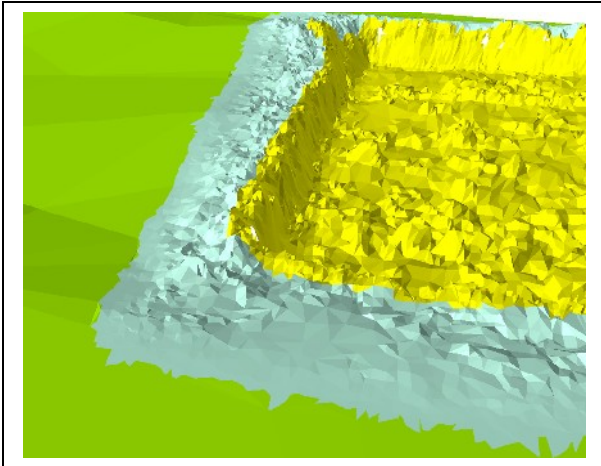


Figure 10. The spline height model is applied: transition surface guarantees now C0 and C1 continuity at the borders.

## 2.4 Mesh simplification

Although topological and geometric continuity have been obtained so far, a transition in terms of point density has not been modelled yet. However, we want density to decrease gradually, as we move from the inner to the outer border of the transition surface, so we need to simplify the mesh accordingly. Several mesh simplification algorithms have been developed in the last 20 years in the field of Computer Graphics (Luebke, Reddy *et al.*, 2002), like for example point decimation, edge collapse, triangle collapse techniques – just to name some –, which permit to eliminate selected points of the mesh, within a certain error value. Among the existing different approaches, “Simplification Envelopes” (Cohen, Varshney *et al.*, 1996) seem to fulfil our needs in terms of topology preservation and varying error value. A variable approximation is needed in fact, in order to preserve details which are not to be simplified beyond a certain level.

Furthermore, point reduction occurs only through simple deletion of selected points, thus no “new” points result from operations typical of other techniques like edge collapse, where out of two edge endpoints a new one is made.

Simplification Envelopes consist of two offset surfaces, which are not more than a (user defined)  $\epsilon$  distant from the original surface. The “outer” surface is created by a displacement along the normal vector of every vertex by  $\epsilon$  and the “inner” surface is created by displacing by  $-\epsilon$ . Since the envelopes are not allowed to self-intersect, the simplified model surface will lie between the offset surfaces. Mesh simplification is obtained iteratively: a mesh vertex is removed, a hole is therefore created which is then triangulated again. If all new triangles do not intersect with the offset surfaces, the point deletion and the new triangles are accepted, thus the algorithm continues with the next vertex.

In general, the bigger the  $\epsilon$  value is, the higher the expected simplification of the mesh will be. Moreover we want a gradually increasing simplification from the inner border to the outer border.

Since the  $\epsilon$  value can be assigned to the input mesh on a per vertex basis, we can set  $\epsilon=0$  for the inner border points (no simplification at all) and let it grow toward the outer border up to a  $\epsilon_{max}$  value.

Like for the weight function  $y=f(x)$  in (1), a proper distance-dependent  $\epsilon$  function has to be shaped (from the inner to the outer border). Thus we define similarly:

$$\epsilon = g(x, \epsilon_{max}) = f(1-x) \cdot \epsilon_{max} \quad (3)$$

where:

$f(1-x)$  = a spline function, as in (1)

$x$  = distance parameter, as in (2)

$\epsilon_{max}$  = maximum displacement value for the offset surfaces

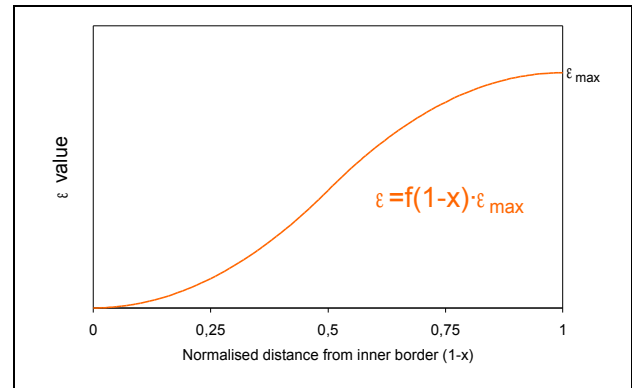


Figure 11. Similarly to the weights for the  $\Delta z$  values, the  $\epsilon$  function is modelled upon a spline basis.

So far, the  $\epsilon_{max}$  value needs to be determined heuristically, but further work is in progress in order to define an approach which can automate this step. In Figure 12 some examples of mesh simplification with growing values of  $\epsilon_{max}$  are shown. In Figure 13 the transition surface between the DTM and the archaeological pit is represented ( $\epsilon_{max}=8$  cm).

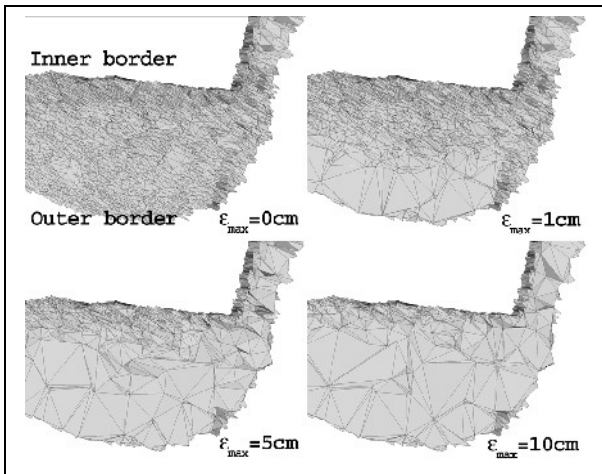


Figure 12. Mesh simplification with growing values of  $\epsilon_{max}$ . ( $\epsilon_{max}=0$  corresponds to the original input mesh, no simplification is performed)

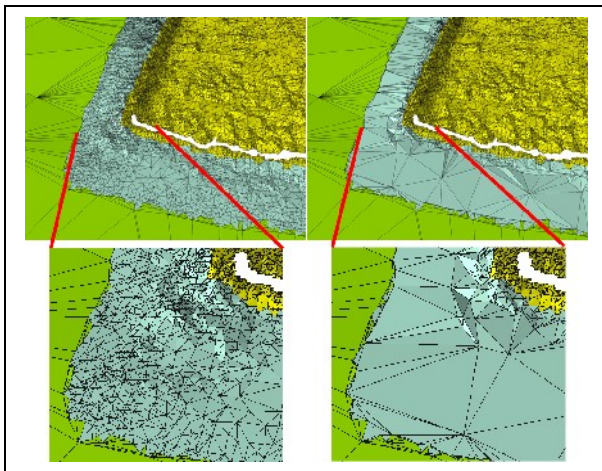


Figure 13. The transition surface, in azure, before (left) and after (right) progressive mesh simplification. The  $\epsilon_{max}$  value chosen is 8 cm.

## 2.5 Implementation

Although adopting open source software was not our primary goal, we have tried to use as much already available and free development tools as possible. Implementation has been carried out mainly in the PostgreSQL object-relational database (version 8.2), in tight conjunction with its “extension” PostGIS (version 1.3), which adds support for geographic objects allowing the PostgreSQL server to be used as a backend spatial database for geographic information systems. Moreover, PostGIS follows the OpenGIS “Simple Features Specification for SQL”.

Server functionality has been extended with one of the built-in procedural languages (PL/pgSQL), which allows for easy query scripting and grouping inside the database server, thus joining the power of a procedural language and the ease of use of SQL, but with considerable savings because there is less client/server communication overhead (otherwise every SQL statement should be executed individually by the database server).

Quantum GIS (version 0.9) was chosen as the visualisation and inspection tool, given its support for the PostgreSQL/PostGIS datasets. Furthermore, it offers an easy integration with GRASS GIS, which was used for 3D views (on GNU/Linux), while

proprietary ESRI ArcScene was the 3D visualisation choice under Windows.

For mesh inspection and (light) editing tasks, we have used MeshLab (version 1.1), a free and open source extensible platform for processing and editing of 3D triangular meshes.

## 3. INITIAL TEST RESULTS

In the following pictures we present some more test results. Input data are meshes originating from datasets of an excavation site. The low resolution model is a DTM whose points were acquired through a stop-and-go GPS surveying campaign. The high resolution meshes refer to excavation pits and have been obtained from terrestrial laser scanner point clouds.

All the meshes (Figure 14) are georeferenced, topologically well-behaving, “clean” (no outliers, no mesh self-intersections, etc) and 2.5D.

With the usual colours used throughout this paper, we identify in the following Figures (15 to 20) the low resolution model with green, the transition surface with azure and high resolution model with yellow.

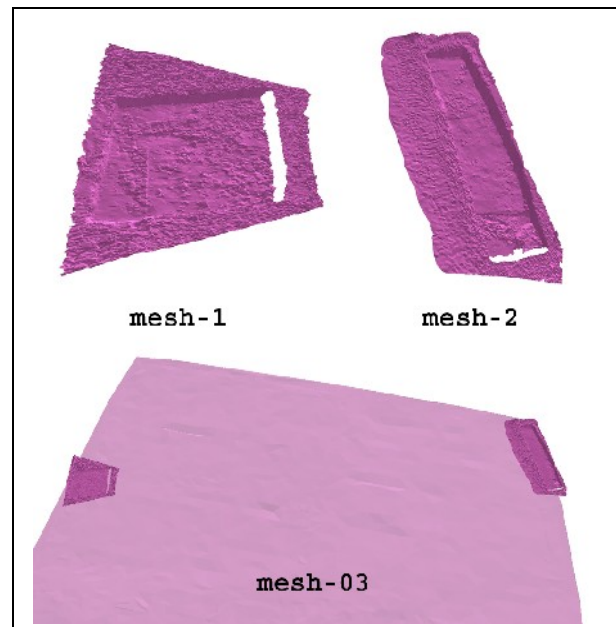


Figure 14. Input models: high resolution mesh-1 and mesh-2 (from laser scanner point clouds) and a global view of the low resolution mesh-3 with the not yet integrated datasets.

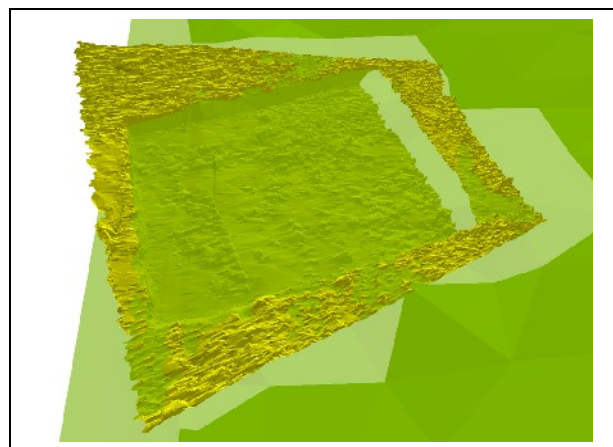


Figure 15. Mesh-1 and mesh-3 overlapped

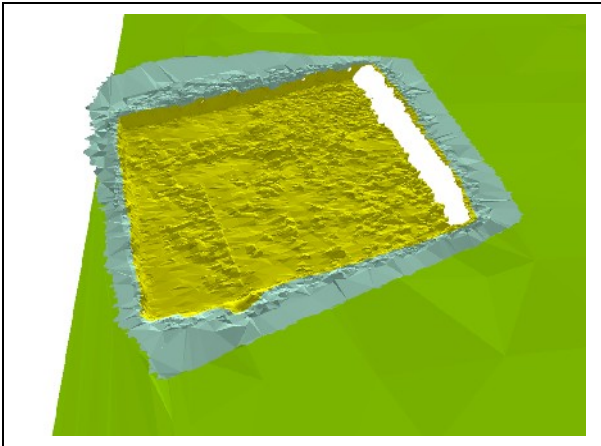


Figure 16. Integration of mesh-1 and mesh-3

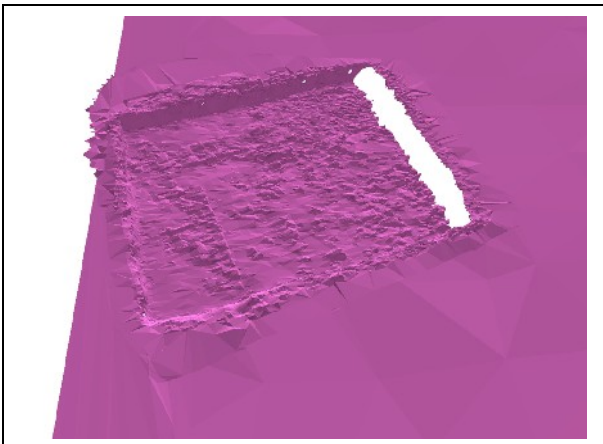


Figure 17. Resulting model obtained from the integration of mesh-1 and mesh-3.

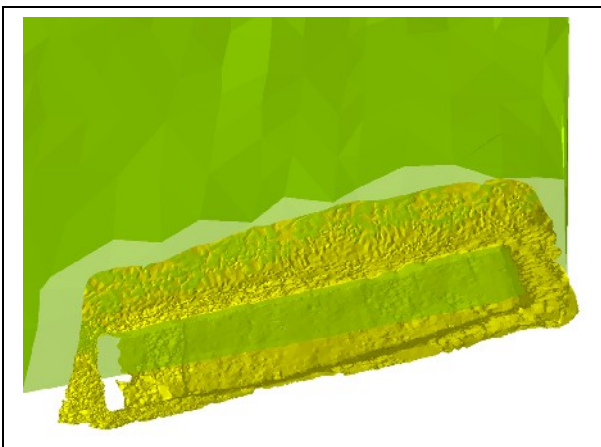


Figure 18. Mesh-2 overlapped with mesh-3

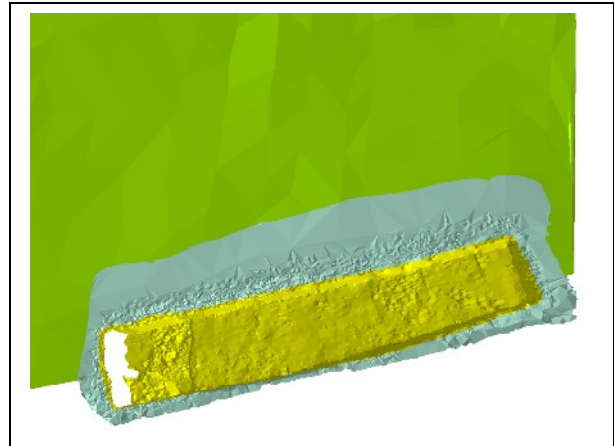


Figure 19. Integration of mesh-2 and mesh-3

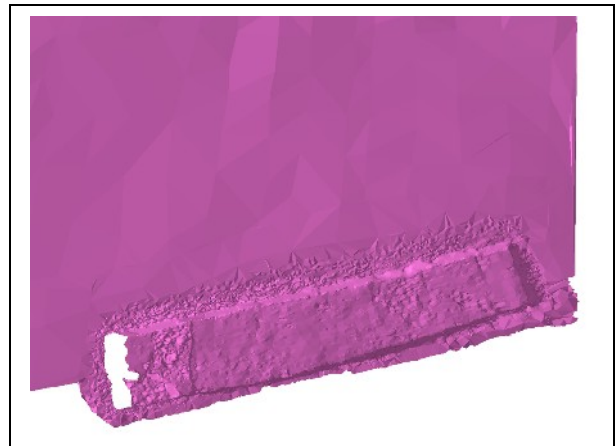


Figure 20. Resulting model obtained from the integration of mesh-2 and mesh-3

#### 4. CONCLUSIONS AND OUTLOOK

In this paper we have presented a method for the creation of a transition surface between a low resolution DTM and a high resolution model. This enables a local update/enhancement of the DTM through a geometrically and topologically correct insertion of a high resolution model. Condition is that we can use some extra information, which we commonly define as “collar”.

Our approach allows for the computation of a (so far deterministic) transition surface that fulfils the conditions and characteristics that were set initially. However some aspects still remain to be implemented/enhanced in order to reduce the number of initial simplification hypotheses.

The following points will be subject of future work:

- Determination of the appropriate size for the overlapping zone and (semi)-automatic identification of the borders (when not given as input).
- Development of a CityGML compliant export of nested DTMs, which also considers information about validity extent polygons and delivers the resulting transition surface in proper format.
- Simplification of the points belonging to the outer border polyline is not yet implemented.
- Distance (height difference) values between the two input meshes are calculated on the z-axis, however this is a limiting factor (2.5D models only). Other approaches are

being considered with the goal of generalisation to 3D models.

- Presently, determination of the border distance parameter  $x$  occurs on the basis of buffers created on the  $xy$ -plane. An improvement should consider calculating geodesic shortest paths on the actual polyhedral surface. Although this might lead to a circular problem, because we might seek to calculate distances upon a yet to obtain surface, this could be solved by an iterative procedure.
- Since the simplification value  $\epsilon_{\max}$  is so far heuristically defined, this implies some amount of arbitrariness. On the other hand, we plan on assigning to it a proper value which accounts for point accuracy and point density.

In the light of the above-named planned improvements, and since any consideration about precision/accuracy has been intentionally omitted in this initial setup phase, we are presently working on a general overhaul of the model through adoption of a stochastic approach, which should help at solving some of the pending issues.

However, while more tests on 2.5D input models are being carried out, results obtained so far with this deterministic approach already provide good integration results with respect to topological correctness and geometric continuity.

## REFERENCES

- Cohen J., Varshney A., Manocha D., Turk G., Weber H., Agarwal P. K., Brooks Jr F. P., Wright W. V., 1996: *Simplification Envelopes*, Proceedings of Siggraph (New Orleans, LA, August 4-9, 1996)
- Gröger G., Kolbe T.H., Czerwinski A., Nagel C., 2008, OpenGIS City Geographic Markup Language (CityGML) Encoding Standard, Version 1.0.0, OGC Doc. No. 08-007r1, Open Geospatial Consortium.
- Kraus K., Karel W., Briese C., Mandlbürger G., 2005, *Local accuracy measures for digital terrain models*. Photogrammetric Record, Blackwell Publishing, Vol. 21 Issue 116, Pages 342-354
- Kraus K., Karel W., 2006, *Quality Parameters of DTM*, in Checking and Improving of Digital Terrain Models/Reliability of Direct Georeferencing, Official Publication No 51, European Spatial Data Research (EuroSDR), ISBN: 9789051794915, Pages 125-139.
- Koch A., 2005, *An Integrated Semantically Correct 2.5D Object Oriented TIN*. Gröger, G., Kolbe, T.H. (eds) 1st Intern. ISPRS/EuroSDR/DGPF-Workshop on Next Generation 3D City Models. Bonn, Germany, EuroSDR Publication No. 49.
- Kolbe T.H., 2008, *Representing and Exchanging 3D City Models with CityGML*, In: Lee/Zlatanova (eds): Proceedings of the 3rd International Workshop on 3D Geo-Information, Seoul, Korea. Lecture Notes in Geoinformation & Cartography, Springer Verlag.
- Laurini R., 1998, *Spatial multi-database topological continuity and indexing: a step towards seamless GIS data interoperability*. Int. Journal on Geographical Inform. Science, Vol. 12, No. 4.
- Luebke D., Reddy M., Cohen J., Varshney A., Watson B., Huebner R., 2002, *Level of Detail for 3D Graphics*, Morgan Kaufmann Publishing, ISBN 1-55860-838-9
- Schmittwilken J., Saatkamp J., Förstner W., Kolbe T.H., Plümer L., 2007, *A Semantic Model of Stairs in Building Collars*, In: Photogrammetrie, Fernerkundung, Geoinformation PFG., S. 415-428.
- Shewchuk J.R., 1996, *Triangle: Engineering a 2D Quality Mesh Generator and Delaunay Triangulator*, in "Applied Computational Geometry: Towards Geometric Engineering" (Ming C. Lin and Dinesh Manocha, editors), Vol. 1148 of Lecture Notes in Computer Science, pg. 203-222, Springer-Verlag, Berlin
- Stadler A., Kolbe T.H., 2007, *Spatio-Semantic Coherence in the Integration of 3D City Models*. In: Proceedings of the 5th International Symposium on Spatial Data Quality ISSDQ 2007, Enschede, Netherlands, ISPRS Archives.

## SOFTWARE RESOURCES:

PostgreSQL: <http://www.postgresql.org>  
PostGIS: <http://postgis.refractor.net>  
MeshLab: <http://meshlab.sourceforge.net>  
Quantum GIS: <http://www.qgis.org/>  
GRASS GIS: <http://grass.itc.it/>  
Triangle: <http://www.cs.cmu.edu/~quake/triangle.html>  
Simplification Envelopes:  
<http://www.cs.unc.edu/~geom/envelope.html>

## ACKNOWLEDGEMENTS

We would like to express our gratitude to both DAAD (Deutscher Akademischer Austausch Dienst) and Fondazione Cassa di Risparmio di Padova e Rovigo who granted scholarships for carrying out part of this research work in Germany and Italy.

Special thanks to Jonathan Cohen and Amitabh Varshney for kindly providing the source code of "Simplification Envelopes". We would like to thank our colleagues Lothar Gründig and Christian Clemen for fruitful discussions.

The datasets presented in this paper have been acquired and partly processed by the members of the Laboratorio di Rilevamento e Geomatica (Università di Padova). Our thanks to all of them.