RELIEF ASPECTS

A METHOD FOR CREATING PAPERCRAFT RAISED RELIEF MAPS FROM DIGITAL ELEVATION MODELS

Jürnjakob Dugge[1], Johann Dugge[2]
[1] Stuttgart, Germany
[2] Brussels, Belgium

ABSTRACT

Raised relief maps provide a particularly intuitive and engaging way to represent topography. Depending on the intended use of the map, different methods are used for producing raised relief maps. These include manual sculpting and painting of plaster models, manual or semi-automatic construction from wood or cardboard layers, and automatic production using a 3D printer. These methods vary in terms of accuracy and realism of the final product, the cost, effort and skill involved in the production, and in the suitability for reproducing different types of topography. In general, the production of raised relief models is either expensive, difficult, or labour intensive.

We present a method for producing low-cost full-colour raised relief maps from digital elevation models (DEMs) by creating a papercraft model of the DEM. The method involves the following steps: converting the DEM to a triangulated irregular network (TIN), unfolding the TIN, printing the unfolded TIN on paper, and assembling the printout into a physical 3D model. This method allows for the production of raised relief maps without the need for

specialised equipment or extensive training and experience. The degree of realism of the resulting maps is similar to that of wood layer models.

The quality of the final raised relief model depends crucially on the characteristics of the TIN. We discuss the requirements that make a TIN suitable for use in a papercraft raised relief model and present a novel algorithm for converting a gridded DEM to a TIN that takes these requirements into account.

Keywords: raised relief map, paper model, triangulated irregular network

1 INTRODUCTION

Raised relief maps, also called terrain models or relief models, are the most immediate way to represent topography. In conventional maps, the elevation information has to be encoded in the form of spot heights, contour lines or relief shading, which requires knowledge and experience in order to be decoded by the viewer. By contrast, raised relief maps, as three-dimensional models of the physical landscape, convey the elevation information directly and can thus be understood by the viewer without special training or explanation. It is this immediate representation of physical features and the provision of a bird's eye view that makes raised relief maps so fascinating and attractive.

Merely looking at a raised relief map already fosters an understanding of the topography that would be hard to achieve with topographic maps alone, but going through the process of constructing a raised relief map does even more to enhance the understanding of the landscape and of the map that represents it: Eduard Imhof gives the advice that "at least once in their early years, cartographers, topographers, geographers, and geologists should construct a terrain model based on an interesting contour plan" (Imhof 1982). Today, digital elevation models have taken the contour plan's place as the method of choice for storing and transmitting elevation information, and so the recommendation might now be that at least once should we turn the digital elevation data we are working with into a physical artefact that we can hold in our hands. In this paper, we present a method for following this advice by creating papercraft raised relief maps from digital elevation models.

A vast array of methods for creating raised relief maps exist, varying in accuracy and realism of the final product, the cost, effort and skill involved in the production, and in the suitability for reproducing different types of topography. The website terrainmodels.com provides a good overview of the different methods and showcases a large number of different raised relief models.

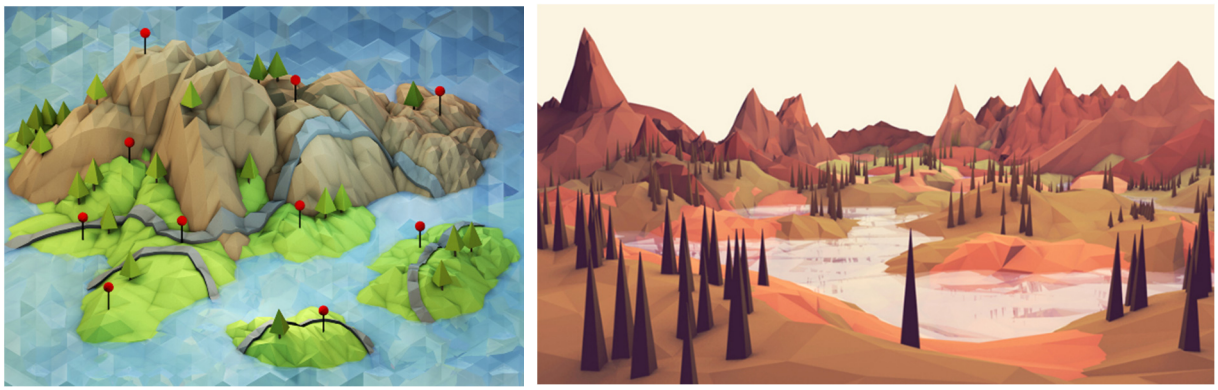
Among the most realistic raised relief maps are hand-made plaster models. Professionally made plaster models are mainly used for touristic and educational purposes in museums, exhibitions and visitor centres (Räber and Hurni 2008), where they often form the centre of attention (Buchroithner 2007). The process of creating such models is very involved and takes years to master (Mair 2012), making it infeasible for many applications.

A much faster and more affordable way of creating detailed raised relief maps that has been gaining traction is the use of 3D printers (e.g. von Wyss 2015). Currently, most consumer grade 3D printers can only produce relatively small uncoloured models.

The simplest and most wide-spread type of raised relief map is the layer model or step model, which consists of a stack of thick paperboard or wood layers, each layer corresponding to a

contour line on a topographic map. These raised relief maps are popular for use in architectural models, because their production is comparatively easy and inexpensive, and because the resulting horizontal surfaces allow for the simple addition of building models. Layer models are mainly useful for depicting gentle terrain types. For areas with steep terrain and several isolated peaks, the number of layers and layer parts that need to be cut, positioned and glued quickly makes the construction cumbersome.

A different approach for creating physical 3D models without specialised equipment is to cut out pieces of paper that can be glued together to form a hollow body in the shape of the object to be modelled, resulting in a "papercraft" model. In order to use this technique for creating a raised relief map, a suitable representation of the landscape surface has to be found that can be cut from paper sheets. One way to generate such a representation is to use a manifold of triangles approximating the surface, similar to the way 3D objects are represented in computer games. Using only a small number of triangles results in a faceted appearance, the "low-poly aesthetic" that is found in the work of visual artists since the early 2010's, where low numbers of polygons are used not due to technical limitations, but as an artistic technique (Schneider 2014). The work of Timothy Reynolds is an example of this visual style where the structure of the underlying 3D models is deliberately made visible (Figure 1).



a) *Map Wars*, 2013. b) *Untitled*, 2014.

Figure 1: Examples of "low-poly aesthetic" artwork by Timothy Reynolds, where the individual polygons of the underlying 3D models are deliberately made visible. Reproduced with permission by the artist.

The process for creating a papercraft raised relief map from a digital elevation model consists of the following steps, which will be described below:

- creating a 3D triangle mesh that approximates the digital elevation model
- optionally adding a texture to the mesh
- unfolding the mesh to a 2D representation
- printing and cutting the unfolded mesh
- assembling the cut out mesh into a physical 3D model.

2 CREATING THE MESH

2.1 MESH PROPERTY REQUIREMENTS

In order for the mesh to be suitable for constructing physical paper models, the triangle elements of the mesh need to fulfill a number of quality criteria.

High fidelity of the approximation: The triangulation should be as faithful to the digital elevation model as possible, that is, the overall error between the approximation and the full gridded terrain model should be as small as possible. There are different ways of measuring the approximation fidelity, with the two most common error metrics used for assessing the quality of a surface approximation being the maximum vertical distance between the grid and the mesh, and the root mean square of the vertical distances (Heckbert and Garland 1997). These metrics have the advantage of being fast to evaluate, but they are overly sensitive to horizontal errors. A more robust method is the sampled symmetric closest distance described in Garland (1999): for every point in the original grid, the distance to the closest point on the grid-sampled triangle mesh is calculated, and conversely for every point on the grid-sampled triangle mesh, the distance to the closest point on the original grid is calculated; since these distances are not generally the same, the average of both values is taken.

Small number of triangles: When constructing the paper model from the triangulation, every triangle requires a certain amount of time: it needs to be scored, cut, folded and glued. The construction process also becomes more complicated with increasing numbers of triangles, since the number of possible combinations rises, and the overall accuracy of the paper model decreases, since inaccuracies in gluing parts can accumulate. Therefore, in order to ensure that the construction process remains possible in practice, the number of triangles in the triangulation needs to be limited. Takahashi et al. (2011) suggest an upper limit of 500 triangles. Generally, the smaller the number of triangles, the faster, easier and more accurate is the construction process.

Similar triangle size throughout the mesh: In triangle approximations, areas of the original mesh that are planar can be represented using only a few large triangles, while strongly undulating areas need to be represented using many small triangles. For most applications, this adaptive nature of triangulated irregular networks is a desirable feature, since it reduces the storage size of the network by eliminating redundant information. However, from an aesthetic point of view, meshes with large variations in triangle size are undesirable. The difference in triangle size draws unwanted attention to the nature of the approximation and distracts from the object being portrayed. While from a mathematical point of view it might not be necessary to split up large planar areas into smaller triangles, it helps the viewer to better understand the landscape features when there is a relatively homogenous "resolution" to the model. A single large triangle next to many small ones tends to look like an error or a patch of missing data; splitting that triangle up into several smaller triangles, while effectively resulting in the same surface, communicates to the viewer more clearly that the planarity of the area is indeed an accurate representation of the underlying data, and not merely an artifact introduced by the triangulation.

Good triangle shapes: The shapes of the triangles have a large impact on the ease of construction of the resulting paper model. Long, narrow "sliver triangles" are problematic for constructing paper models for several reasons: they are more sensitive to inaccuracies when cutting or scoring the triangle edge, as a small angular deviation from the correct direction results in a large relative error of the resulting triangle's area and edge lengths; they are harder

to fold along the edge, since the narrow tips have a tendency to bend instead of fold; and they offer only small area for attaching the glue tab of a neighbouring triangle, making the connection less secure. The ideal triangle element for our application is an equilateral triangle. Deviations from this ideal shape can be measured in a number of ways (Shewchuk, 2002), one useful metric being the shape regularity quality q proposed by Bank and Smith (1997): for each triangle, we calculate the ratio of the triangle area and the sum of the squared edge lengths, normalised to equal one for an equilateral triangle and approach zero for a degenerate triangle.

2.2 PREVIOUS ALGORITHMS

A vast number of algorithms have been proposed for generating polygonal meshes from high resolution data. For good overviews of the topic, see Heckbert and Garland (1997) and Luebke (2001).

For the application of creating papercraft raised relief maps, we considered a number of existing algorithms. We briefly describe the algorithms, and show the result generated when applying them to the same elevation dataset of the Matterhorn, specifying a target triangle count of 150. The elevation dataset has a resolution of 400 x 400 pixels, covering an area of 2,000 m x 2,000 m (Figure 2a).

The most straightforward way of triangulating a gridded elevation dataset is to create a downsampled version of the original grid and then create a regular grid triangulation (Figure 2b). Generally, this algorithm leads to a poor approximation fidelity, because there is no mechanism for including important characteristic points of the terrain in the triangulation, so peaks, valleys and ridges are easily missed.

In order to improve the approximation fidelity, the points to be included in the triangulation should be chosen not by using regular sampling, but based on their contribution to the overall terrain shape.

One popular method for choosing the points is the greedy refinement algorithm described by Garland and Heckbert (1995). It starts with a triangulation consisting of the four corner points of the domain. In each step, the vertical errors between each grid point and the triangulation are calculated, and the point with the largest vertical error is added to the mesh (Figure 2c). This algorithm is extremely fast, but because it only regards the largest vertical error it does not typically lead to results with low overall errors, and because it does not take the shapes of the resulting triangles into account when choosing which points to add, the resulting triangulation often includes many sliver triangles and triangles of uneven sizes.

Instead of iteratively refining a simple mesh, decimation algorithms start with a full triangulation and iteratively simplify the model. The popular QSLIM algorithm (Heckbert and Garland 1999) simplifies the model by merging the vertices of edges, starting with the merge operation that has the smallest impact on the overall error. This algorithm typically performs much better than the greedy refinement algorithm in terms of approximation fidelity and triangle quality, but still misses some characteristic features of the terrain and includes narrow triangles (Figure 2d).

None of the surveyed methods were found suitable for the specific purpose of generating meshes for building papercraft models. We therefore propose a new method using an optimisation approach that takes into account all of the requirements described in section 2.1.

2.3 PROPOSED OPTIMISATION ALGORITHM

In an optimisation algorithm, an initial solution to a given problem is iteratively improved by making small adjustments, evaluating the quality of the adjusted solution according to some objective function, and using the new solution as the starting point for the next iteration, approaching an optimum solution over time.

The objective function forms the heart of any optimisation algorithm. It is important that it captures the essence of the problem, but approximations may be necessary to reduce the computational cost of evaluating potentially thousands of solutions. In our proposed method, we use an objective function, described below, that evaluates the approximation fidelity and the quality of the triangle shapes. The number of triangles is specified by the user, and the desired similarity in triangle sizes is implicitly accounted for by the triangle shapes and by the initial solution.

Our method starts by generating a mesh with a specified number of triangles using a greedy algorithm (Stage 1) and then attempts to improve it iteratively by repeatedly adjusting either the mesh connectivity (Stage 2a) or shifting node positions (Stage 2b). Throughout stage 2, the number of triangles as well as the number of nodes on the boundary and inside of the DEM domain remains unchanged. In each iteration, changes to node position or connectivity are evaluated by this objective function and accepted only if they yield an improvement.

2.4 STAGES

Stage 1: Initial Vertex Distribution

Areas of large variation in the terrain will require a higher relative density of nodes than flat areas. The initial vertex placement is critical to the final result of our optimisation algorithm because it will only proceed with changes if they bring an improvement to the objective function M_T and it is difficult to escape from a local optimum.

Our meshing starts with four points placed on the corners of the domain, yielding two triangles. For each additional point P_B on the boundary we gain one additional triangle while each point P_I on the interior of the domain yields two additional triangles. The number n of triangles can be computed as $n = 2 + (P_B - 4) + 2\ P_I$ (de Berg 2008: 193).

As we have two independent variables, many solutions may be possible to obtain a certain number n of triangles. We introduce a new constraint: assuming all triangles had the same size, each would cover an area $A_T = A_D/n$, where A_D is the total area of the domain. The edge length a_T of an equilateral triangle with area A_T is given by $a_T = \sqrt{4\,A_T/\sqrt{3}}$. Dividing the boundary of the domain a_D into segments of length a_T gives the number of nodes to be placed on the boundary, P_B. The number of nodes to be placed inside can now be calculated as P_I= round(½ (n + 2 - P_B)). If needed for achieving the specified number of triangles, one additional node is added to the boundary.

a) Source DEM

b) Regular

c) Greedy

d) QSLIM

e) Proposed method

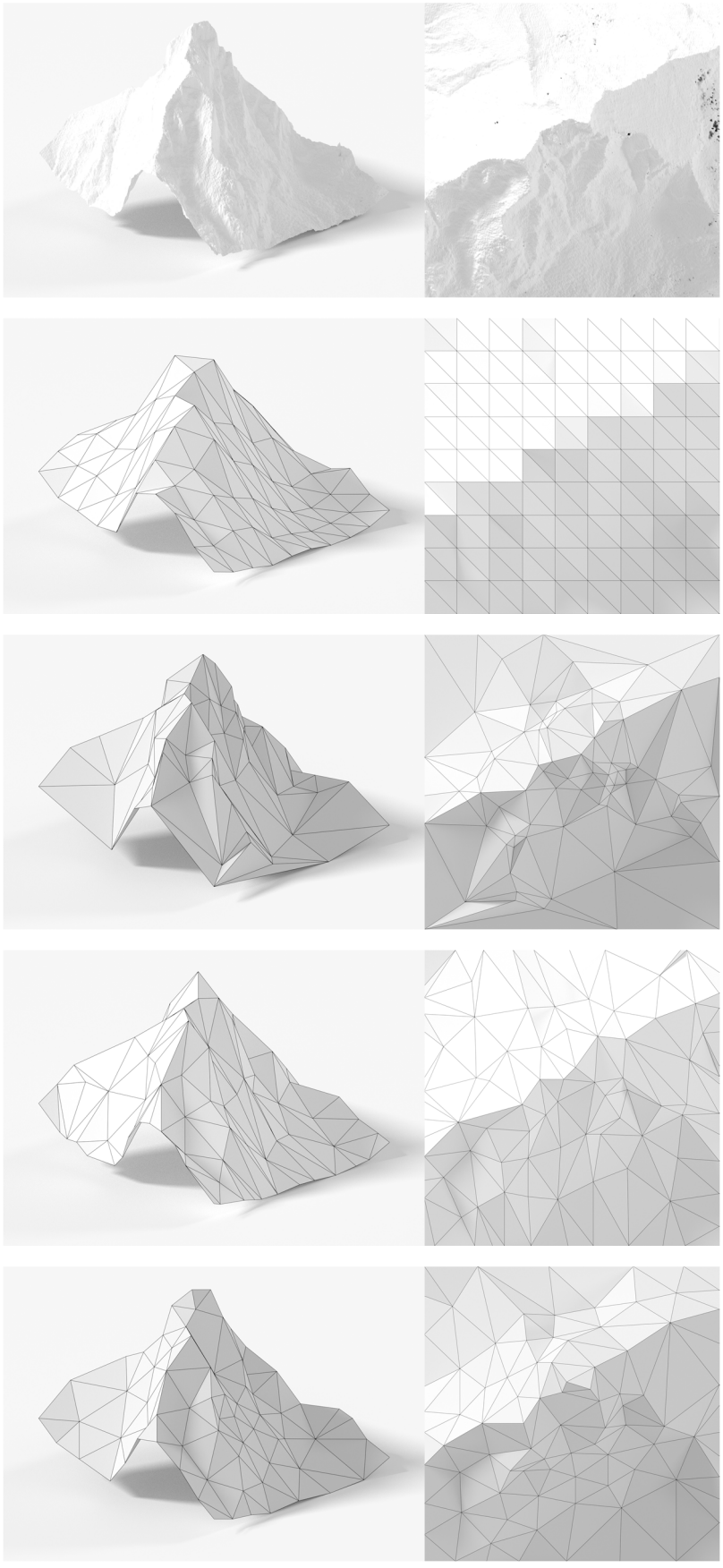


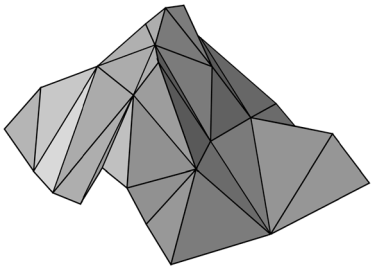
Figure 2: Comparison of the triangle meshes generated using different existing algorithms and the proposed method, showing an oblique and a top-down view.

To distribute the additional points, a method similar to the greedy algorithm described in section 2.2 is used. However, instead of using just the vertical error for determining which point to add, we first weight the error using the maximum local absolute Gaussian curvature to identify important features such as peaks, ridges and gullies, as described by Leonowicz (2010). The search for the point with the largest vertical error is constrained to the domain boundary until all P_B boundary points have been added. The internal points P_I are then distributed in a similar fashion, however this time excluding the domain boundary from the candidate search. This two-stage approach reduces the risk of creating sliver triangles along the domain boundary.

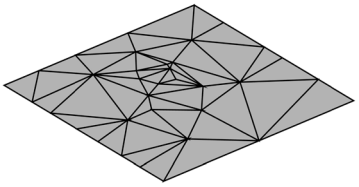
Objective Function: We define two competing measures for our objective: the average terrain fidelity and the average triangle shape quality.

For calculating the terrain fidelity M_F, we use the average vertical error over all grid points, normalized by the variance of the terrain grid, and weighted using the inverse of the horizontal distance from the point to the edge of the corresponding triangle. The distance is measured as the minimum of the barycentric coordinates of the point raised to a power p=2, plus 1, then scaled to a mean of 1, which assigns greater weight to points on the triangle edges, since these have a greater impact on the appearance of the model (Figure 3).

a) Oblique view of a triangle mesh



b) 2D projection of the mesh



c) Weight map for the the vertical errors. Errors on triangle edges are weighted more strongly than errors in triangle interiors

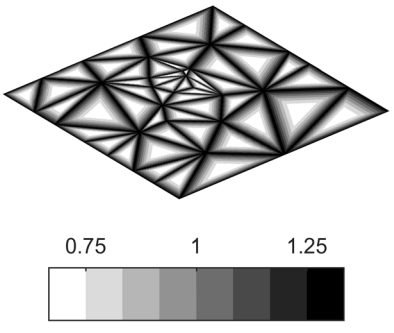


Figure 3: Vertical error weighting.

The triangle shape quality q described in section 2.1 is averaged across all triangles to yield an overall shape metric M_S, with all triangles contributing equally to the overall score, regardless of their area. By penalising triangles with small angles, large variations in triangle area between neighbouring triangles are implicitly penalised as well.

To combine the fidelity and shape metrics into an overall objective function, a weighted average M_T for the whole triangulation is computed as $M_T = c\ M_F + (1-c)\ M_S$ Values of c=0.8-0.95 have yielded good results in our experiments.

Stage 2a: Quad Flipping

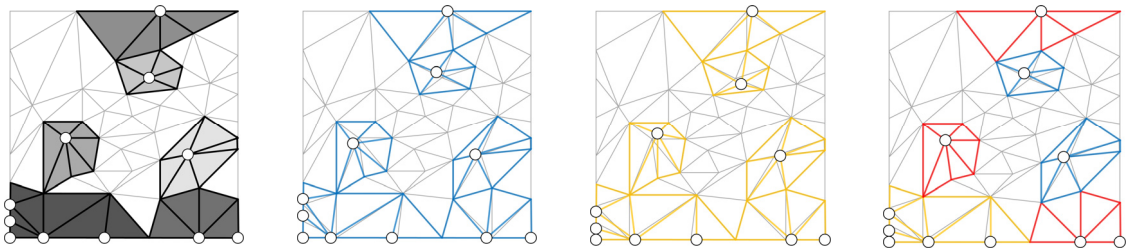
For any pair of adjacent triangles (quads) that form a convex boundary in 2D there are two alternative connectivities where the triangles do not overlap. Unless all points are in the same plane, these yield different surfaces – one with a ridge, and the other with a valley. Our first step in each iteration is to check if flipping any quads yields an improvement in the objective function. Quads that rank the poorest in contribution to the objective function will be checked first. After a flip all quads of the new mesh are ranked again and the process is repeated until no further improvement can be obtained through flipping.

Stage 2b: Node Shifting

After the quad flipping stage, the positions of a randomly selected set of nodes are slightly shifted horizontally and assigned the elevation values of the gridded dataset at the corresponding new position. For shifting the node positions, we used a method based on the concept of *Simultaneous Perturbation Stochastic Approximation* described by Spall (1998): in each step a perturbation of the mesh is performed in opposite directions, and the direction yielding the best improvement of the total objective M_T is selected. If neither perturbation improves the objective another random perturbation is generated (Figure 4).

In our implementation, when a certain number of unsuccessful perturbations has been reached we reduce the number of randomly selected nodes that are perturbed, which allows discovery of more localised improvements as the groups become smaller. The optimisation ends when the number of perturbed nodes falls below a certain threshold.

Perturbations can be larger than the length of a single grid cell, which makes it possible to overcome local minima by skipping some cells. To improve convergence, the nodes' perturbations are grouped into independent areas, and each group is evaluated and improved separately.



a) Input mesh and 11 randomly selected nodes, forming 6 groups.

b) Random perturbation of the selected nodes in '+' direction.

c) Opposite perturbation, designated '–' direction.

d) Combination of best '+' / '–' perturbations of the individual groups. Red groups did not improve with perturbation.

Figure 4: *Simultaneous Perturbation Stochastic Approximation* of 11 randomly selected nodes forming 6 groups. Constraints are enforced on edge and corner nodes. Perturbations that would change the orientation of a triangle are dismissed. Notice that the perturbation at each node can have a different magnitude and direction, independent of the others.

2.5 COMPARISON OF RESULTS

From visual inspection of the meshes shown in Figure 2, one can see that the meshes resulting from the different algorithms have very different characteristics. These differences are also evident when analysing the quality criteria set out in section 2.1.

The approximation fidelity metrics of the meshes generated using the different algorithms are shown in Table 1. The proposed method performs best in terms of all error metrics apart from the maximum vertical distance, where the greedy algorithm performed slightly better. This is to be expected, since the objective function of the proposed method does not take the maximum vertical error into account, while the greedy method uses the maximum vertical distance as the sole driving factor. Generally, the average metrics (mean absolute error and root mean squared error) are more meaningful for assessing the overall shape fidelity, with the maximum error primarily being used as an indicator for the presence of outliers.

Table 1: Error metrics for the meshes generated using different algorithms: maximum error, mean absolute error (MAE), root mean squared error (RMSE)

	Vertical distance (m)			Symmetric distance (m)		
Algorithm	Max	MAE	RMSE	Max	MAE	RMSE
Regular	339.9	53.9	70.3	110.2	29.8	35.8
Greedy	199.8	26.8	34.9	86.4	17.0	22.0
QSLIM	215.4	25.7	34.4	77.4	15.4	19.3
Proposed method	202.7	19.4	25.6	53.9	11.9	14.4

To assess the triangle shape quality and the similarity in triangle sizes of the different meshes, all triangles of each mesh are plotted in Figure 5. In the left column, the unscaled triangles are shown: the range of triangle sizes of the regular method and the proposed method are similar, while both the greedy and QSLIM algorithms generate triangles with a much wider range of sizes. The shapes of the triangles can be assessed by normalising the triangles so their widest side has a length of 1 and comparing them to an ideal equilateral triangle, shown in the right column. It can be seen that the triangles generated using the proposed method are generally more similar to the ideal triangle, and don't vary in shape as much as those generated using the other methods.

3 TEXTURING THE MESH

The paper model can either be left uncoloured, or a texture can be applied before unfolding and printing the mesh. Since the model does not contain any overhangs or vertical faces, an image such as a topographic map, a satellite image or an orthophoto can simply be vertically projected onto the mesh. It should be noted that in steep areas of the model, the image is stretched, so the image needs to be of a higher resolution than would be necessary for strictly

vertical viewing. This problem could be mitigated by using terrestrial imagery for the steep areas, or by using a vector topographic map instead of a raster image.

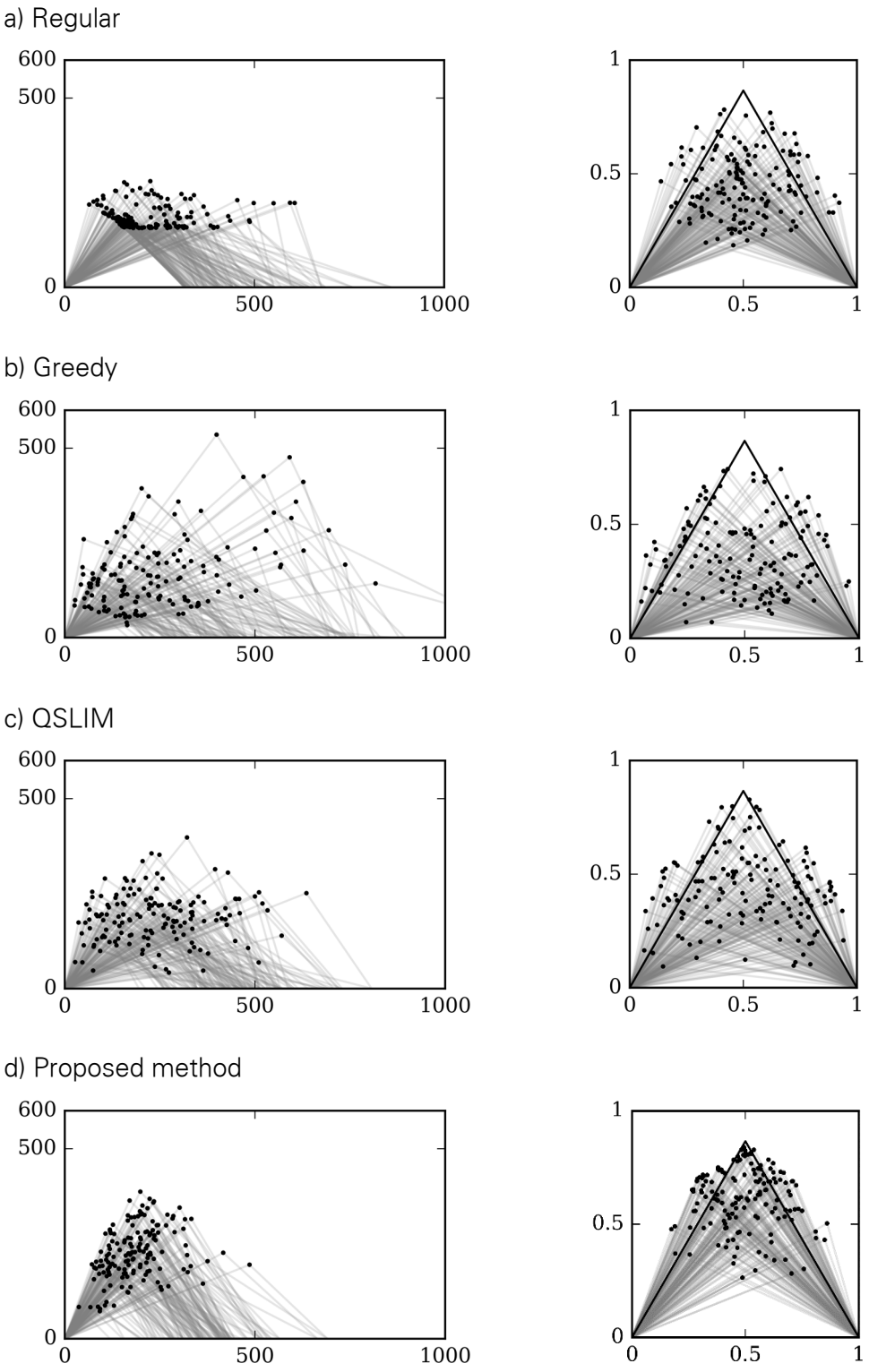


Figure 5: Comparison of the triangle shapes of the triangulations generated using different existing algorithms and the proposed method. Triangles are rotated to have their longest side aligned with the horizontal axis. In the left column, the triangles are unscaled, in the right column, each triangle is scaled so its longest side has a length of 1. For orientation, in the right column an equilateral triangle is plotted in black.

In our experience, satellite or aerial imagery tends to work better for texturing the model than topographic maps: due to the geometry simplification, contour lines of topographic maps will generally not lie strictly at the same level when projected onto the mesh, which can be confusing to the viewer. Similarly, for models with sharp peaks and ridge lines, satellite or aerial imagery taken under strongly oblique lighting conditions is less suitable, because it tends to make the geometry simplification stand out along edges with changes in incident light angles.

4 UNFOLDING THE MESH

In order to print the model, it needs to be unfolded so all the triangles lie in the same plane. To make the model unfoldable, some edges of the mesh need to be cut in such a way that the triangles are only joined along a spanning tree of the original mesh and none of the unfolded triangles overlap (Straub 2011). For the latter requirement, it can be necessary to split the model into separate groups of triangles, so-called islands, that are not connected to each other. Splitting an island further is also necessary if it is too large to fit on a single paper sheet for printing. A naive approach would be to simply cut all edges, resulting in a set of islands each only containing one triangle. However, after printing and folding the model, the cut edges need to be glued back together, which is the most labour intensive part of the model building process. It is therefore desirable not to introduce an unnecessarily high number of cuts. Takahashi et al. (2011) presented a genetic algorithm that attempts to find an unfolding that results in a single island containing all triangles. While this tends to reduce the number of cut-and-glued edges, it also results in a rather unwieldy cut-out that is hard to assemble. We found island sizes of 5 to 20 triangles to generally be the easiest to work with. To unfold the models, we used the "Export paper model" software (Dominec 2016), which prioritises which edges to cut according to a weighted score consisting of the edge length and the angle between the two triangles. The software also automatically generates labelled glue tabs for assembling the model. Some manual editing of the automatically placed cuts can be helpful in order to obtain island shapes that can be more economically arranged on the pages for printing, but in general we found the algorithm to produce satisfactory results without manual intervention.

5 PRINTING, CUTTING AND GLUING THE MESH

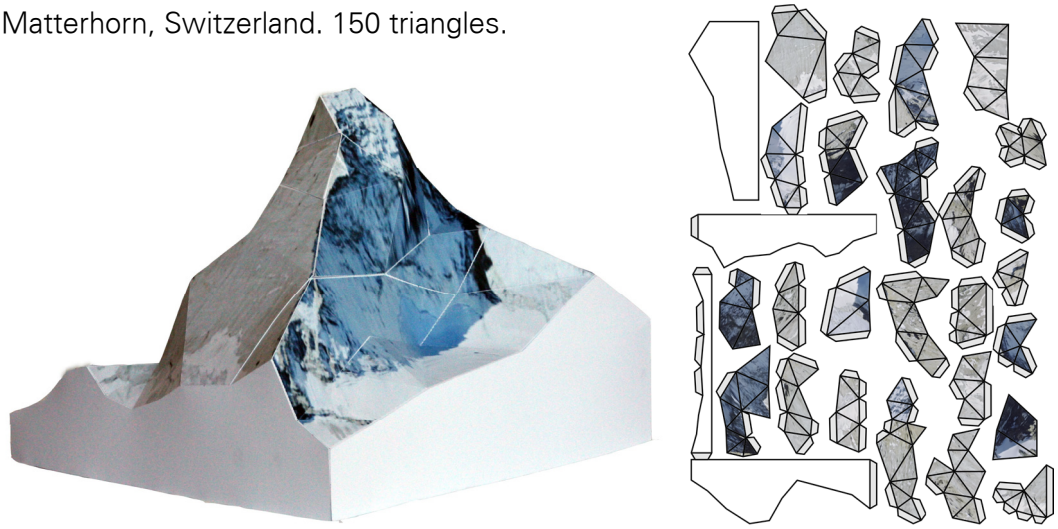
The unfolded islands could be printed on standard office paper, but depending on the size of the final model, stronger paper with a weight of 120 g/m^2 tends to lead to better results. After printing, the islands are cut out and the folding edges are scored in order to obtain a more precise fold. Cutting and scoring can be done manually using a craft knife and a metal straightedge. There are also consumer grade cutting plotters available which can automate the cutting and scoring to some degree. Models which can align the cutting paths with the printed pattern are available for under EUR 200.

After cutting and scoring the islands, the model is folded and the cut edges are glued together. For this step, we found it helpful to have the 3D model available on a computer screen in order to check how the parts need to be folded and joined. Generally, it is advisable to start the assembly process in model areas with sharp folds, as these are harder to assemble at the end.

6 RESULTS

The method described above was used to create papercraft raised relief maps of several topographic features. Figure 6 shows some of the completed models and the unfolded triangles meshes used for constructing the models. The parameters used in the algorithm for converting the gridded elevation data to triangle meshes were kept identical for all three models, only the number of triangles varied.

a) Matterhorn, Switzerland. 150 triangles.



b) Uluru, Australia. 200 triangles.

c) Tre Cime di Lavadero, Italy. 200 triangles.

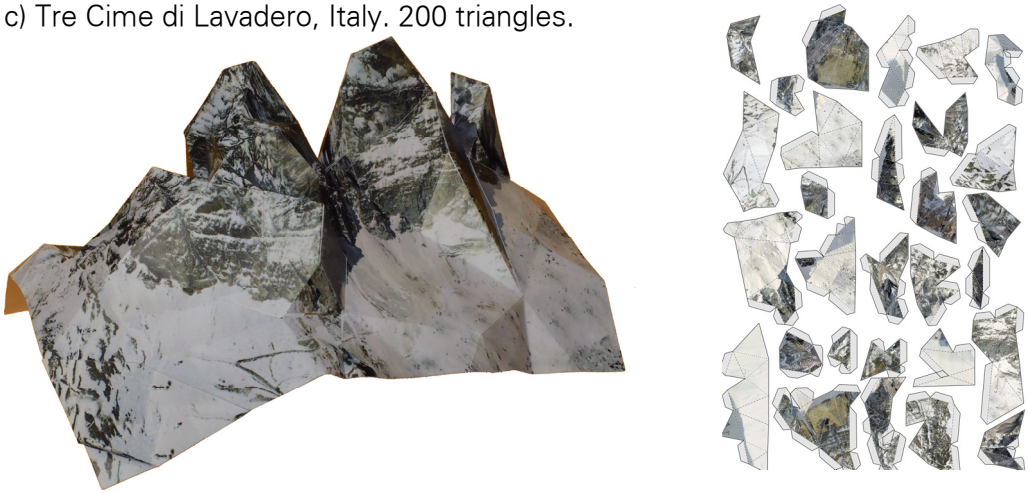


Figure 6: Examples of finished papercraft raised relief models (left column) and the unfolded triangle meshes used for their construction (right column).

7 CONCLUSION

Raised relief maps are a highly intuitive way to represent topography, and making a model is an instructive way to engage with the object being modelled. The presented approach is a practical way to quickly generate low-cost, full-colour models that can be built without requiring intensive training and do not require specialised equipment. The proposed algorithm for converting gridded elevation data to triangulated meshes suitable for papercraft modelling addresses the specific requirements of this application, and produces useful results for a variety of terrain types.

REFERENCES

Bank RE and Smith RK, 1997, Mesh smoothing using a posteriori error estimates, *SIAM Journal on Numerical Analysis*, 34, 979-997.

Buchroithner M, 2007, Echtdreidimensionalität in der Kartographie: Gestern, heute, morgen, *Kartographische Nachrichten,* 57, 239-248.

de Berg M, Cheong O, van Kreveld M and Overmars M, 2008, *Computational Geometry,* Springer, Berlin, Heidelberg, 193.

Dominec A, 2016, Export Paper Model, https://wiki.blender.org/index.php/Extensions:2.6/Py/ Scripts/Import-Export/Paper_Model (accessed July 1, 2016).

Garland M and Heckbert PS, 1995, *Fast Polygonal Approximation of Terrains and Height Fields*, Carnegie Mellon University.

Garland M, 1999, *Quadric-based polygonal surface simplification*, Carnegie Mellon University.

Heckbert PS and Garland M, 1997, Survey of polygonal surface simplification algorithms, Carnegie-Mellon University Technical Report.

Heckbert P S and Garland M, 1999, Optimal triangulation and quadric-based surface simplification, *Computational Geometry,* 14, 49-65.

Imhof E, 1982, *Cartographic relief presentation,* De Gruyter, Berlin, New York.

Leonowicz AM, Jenny B and Hurni L, 2010, Automated Reduction of Visual Complexity in Small-Scale Relief Shading, *Cartographica: The International Journal for Geographic Information and Geovisualization*, 45(1), 64-74.

Luebke DP, 2001, A developer's survey of polygonal simplification algorithms *IEEE Computer Graphics and Applications,* 21(3), 24–35.

Mair T, 2012, The Landscape Relief Model - An anachronism or still a useful object for contemplating the landscape, in: Buchroithner M (ed), *True-3D in Cartography*, Springer, Heidelberg, Berlin, 415-434.

Räber S and Hurni L, 2008, An ambitious Relief Model Project, Mountain Mapping and Visualisation, *Proceedings of the 6th ICA Mountain Cartography Workshop*, 11-15 February 2008, Lenk, Switzerland, 183–193.

Schneider T, 2014, *A Comprehensive History of Low-Poly Art*, https://killscreen.com/articles/poly-generational/ (accessed July 31, 2016).

Shewchuk JR, 2002, What is a Good Linear Finite Element? Interpolation, Conditioning, Anisotropy, and Quality Measures, *Proceedings of the 11th International Meshing Roundtable*, 15-18 September 2002, Ithaca, New York.

Spall JC, 1998, An overview of the simultaneous perturbation method for efficient optimization, *Johns Hopkins APL Technical Digest*, 19(4), 482-492.