Terrain Generalization with Multi-scale Pyramids Constrained by Curvature

Bernhard Jenny, Helen Jenny and Lorenz Hurni

ABSTRACT: This paper presents a method for terrain generalization using Laplacian pyramids. The method pre-processes digital terrain for generating cartographically generalized 3D maps. Map authors can attenuated or amplify selected frequency bands of the terrain with a graphical interface imitating an audio equalizer. Ridge lines and valley lines are localized by curvature indices, and their characteristic shapes are preserved or emphasized using additional sets of equalizer controls. Frequency bands are adjusted separately for the foreground and the background to remove visually disturbing terrain detail in the background. This is relevant for 3D maps in central perspective projection that considerably compresses distant terrain features. The proposed generalization method was implemented in Terrain Equalizer, a free and open-source application providing a graphical interface with interactive 3D previews (available at http://www.terraincartography.com/). Using this application, disturbing high frequency details can be easily removed and major mountain forms can be accentuated. The level of generalization can be adjusted seamlessly from the foreground to the background of the 3D map. Topographic break lines, such as ridge or valley lines, are successfully preserved, which is important for conveying the characteristic shape of a generalized terrain.

KEYWORDS: Terrain generalization, DEM frequency filtering, laplacian pyramid, terrain equalizer

Introduction

Expert cartographers generalize terrain for 3D maps by removing unnecessary and visually distracting details, and accentuating important landforms, while preserving characteristic terrain features and typical landforms. However, generalization methods for 3D maps are most often technology driven, i.e., they are often not targeted at cartographic generalization, but intend to reduce the amount of terrain details in order to achieve responsive frame rates when rendering terrain in video games and other interactive environments. Such methods may introduce artifacts, such as artificial edges and flat triangle structures, or overly smoothed mountain ridges and valleys.

3D maps most often depict terrain in a central perspective projection, compressing terrain features in the background. The terrain compression increases with the distance to the viewer: the foreground is depicted at a large scale, while the background is rendered at a considerably smaller scale. This results in an excessively detailed background, especially when using high-resolution terrains. Such highly detailed representations are often visually inefficient because the main landforms are not clearly discernable due to the many distracting details. The dense details generate disturbing visual noise, which obscures macro topography—it is impossible to see the forest for the trees. Hence, the level of terrain generalization must seamlessly increase with the distance to the viewer.

Cartographic generalization is an inherently visual task. A graphical environment is required for adjusting the generalization parameters in a trial-and-error approach to the spatial resolution of the terrain, the landscape features and the purpose of the map. A problem plaguing authors of 3D maps, however, is the currently available software for generalizing terrain, which sometimes is difficult to comprehend and control, or does not offer a WYSIWYG preview mode for interactive manipulation in real-time.

The research presented in this article aims at making a contribution to the solution of these problems. Visually disturbing details are to be removed from digital terrain, while sharp edges and mountain ridges are to be retained. The goal is to preserve the characteristic appearance of a terrain, and to seamlessly adjust the amount of generalization from foreground to background. We are aiming at a generalization method that is easy to comprehend and control by authors of 3D maps, and provides feedback in real time.

For our prototype, an equalizer metaphor was chosen as basis for the user interface. In audio engineering, the equalizer filter is used to alter the frequency response of an audio source. Most users should be familiar with equalizers from audio

Bernhard Jenny, Helen Jenny, Lorenz Hurni, Institute of Cartography, ETH Zurich, Switzerland, E-mail:<{bernhard.jenny, helen.jenny, lorenz.hurni}@karto.baug.ethz.ch.>
DOI: 10.1559/15230406382110

playback software and physical audio equipment. Terrain Equalizer, a free and open-source software application, was developed, integrating the presented generalization method and its related user interface.

Related Work

In this overview of related research, we concentrate on cartographic generalization aiming at improving the visual effectiveness to generate clear and legible maps. Generalization methods to improve data storage or to accelerate rendering, analysis or transmission of terrain data are not treated here, neither is the generalization of representations derived from terrain models, such as contour lines.

Filtering, Smoothing and Denoising

The term filtering is understood in the context of signal processing, i.e., the application of an operator (the filter) that removes frequency components from the signal (the terrain). Such filters are known from image processing, and can be applied to gridded terrain models. Low-pass filters remove high-frequency details, whereas high-pass filters emphasize discontinuities. While high-pass filters are rarely applied to digital terrain (Weibel and Heller 1991), low-pass filters are the simplest and thus most commonly used filters (Li 2008). Unfortunately, an unintended consequence of isotropic low-pass filters is that smoothing is applied uniformly. Sharp edges and other fundamental features of a terrain are blurred away by the uniform filter. However, if prominent topographic breaks are involved, cartographic practice tells us to preserve their character and not simply smooth them.

The applicability of global smoothing or low-pass filtering, therefore, is limited to smooth rolling terrain and minor scale reductions (Weibel 1992). Alternative anisotropic operators have been studied extensively in image processing, where a wealth of methods are available. Many researchers have been extending image processing techniques to general purpose surface smoothing, aiming at better preserving edges, while reducing noise in the signal. For elevation data, specialized anisotropic smoothing methods have been proposed. Desbrun et al. (2000) present a method for denoising terrain models and other bivariate data based on anisotropic diffusion approaches in image processing. They use a combination of surface area minimization, graph flow, and nonlinear edge-preservation metrics to smooth a two-dimensional dataset, while preserving important discontinuities.

Tasdizen and Whitaker (2003) propose a two-steps method, also aiming at feature preserving smoothing of terrain data, while enhancing discontinuities. Their method forces surfaces towards piece-wise smoothness by first smoothing the field of terrain normal vectors (using a penalty function of curvature), and then fitting a new terrain model to the smoothed normals. Both steps use a gradient descent minimization method with a second-order partial differential equation. Hence, the major shortcoming of this method is the computational complexity and the increased computation time. Hofer et al. (2006) and Mohan et al. (2006) extend these energy minimizing techniques with the concept of tolerance cylinders. These cylinders are derived for each terrain vertex from absolute errors. A cylinder's height represents the absolute vertical accuracy, while the base diameter corresponds to horizontal accuracy. The cylinders are used as hard constraints by the energy minimizing technique, forcing the smoothed terrain surface inside the tolerance bounds defined by the cylinders. Both works aim at respecting the absolute vertical and horizontal errors as specified by the data providers (such as for the DTED or SRTM elevation models). As such, these methods are suitable for denoising terrain by removing outliers, while keeping the surface inside the tolerance bounds. For a more interactive approach to terrain generalization, tolerance cylinders could be locally adjusted, however, the computation time are rather long for both methods due to the computationally intensive energy minimizing techniques.

Wavelet Transforms

Relatively few attempts have so far applied the wavelet transform on digital terrain (Gallant and Hutchinson 1996). Wavelet applications on DEMs are mainly used for feature extraction (Amgaa 2003; Kalbermatten 2010) and terrain simplification by filtering the wavelet coefficients (Mahler 2001; Bjorke and Nilsen 2003). Attempts at using the related Fourier transform go back to the 1970s, but have been found to be less adequate for terrain data (for a review see Kalbermatten 2010, p. 47).

Skeleton Lines

Weibel (1992) proposed the generalization of terrain based on structure lines that are extracted from the terrain model, in some cases using techniques developed for line generalization. Weibel (1992) proposes to use the drainage network and the ridge network, both extracted from the terrain model, depending on the characteristics of the terrain. Many automatic methods have been proposed to characterize terrain patches, extract drainage networks, and break lines. They can be used as input to steer generalization based on skeleton lines, or be applied to morphometric analysis, for example, to selectively remove valleys (Jordan, 2007).

Multi-resolution Terrain

Research in computer graphics has resulted in a variety of methods for reducing the level of detail (LOD) of terrain models. Many of those methods remove terrain details from the background of the scene, aiming at achieving interactive frame rates in games and other applications with interactive terrain visualization. In such applications the main goal is the minimization of data storage and throughput, which leads to the application of TIN structures or nested regular grids at varying resolution (for an overview see Pajarola and Gobbetti 2007). The goal, however, is a purely technical one: The amount of triangles rendered by the computer graphics hardware is reduced, such that visual artifacts are minimized and an interactive frame rate is achieved. This goal is fundamentally different from cartographic generalization for 3D maps, where irrelevant and disturbing micro topography has to be removed, and characteristic terrain structures need to be accentuated.

Multi-resolution TINs can be constructed using various methods, for example, the line simplification method by Douglas and Peucker applied to terrain (Fei and He, 2009), or techniques inspired by Morse functions (Danovaro et al. 2006). Besides terrain visualization, multi-resolution terrain is also applied to filtering LIDAR data (Silván-Cárdenas and Wang 2006).

Frequency Manipulation with Laplacian Pyramids

The main idea of our method presented in this paper is to consider a terrain model as a two-dimensional input signal, where selected frequencies are enhanced or attenuated. A Laplacian pyramid is used to separate consecutive frequency bands. Individual frequency bands are amplified or attenuated to be combined into a new terrain synthesized from the weighted frequency bands.

Laplacian pyramids are commonly used in computer graphics for image blurring, image mosaicing, 3D texturing or for applications in computer vision. The construction and application of Laplacian pyramids for imaging was pioneered by Burt (Burt 1981; Burt and Adelson 1983a; 1983b). The construction of Laplacian pyramids is identical for raster images and raster terrain models. Only the data type used by the algorithm varies, i.e., 8 bit or 16 bit integer numbers for each image channel vs. floating point values for gridded terrain models.

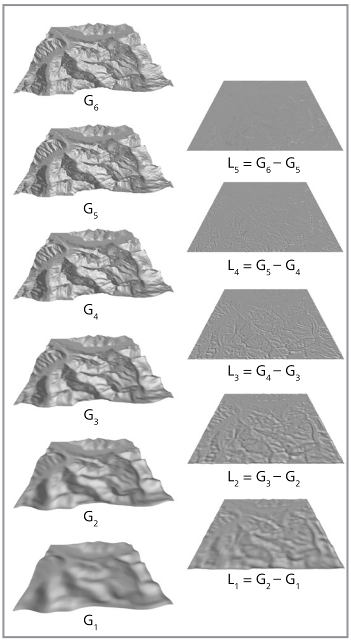


Figure 1. Gaussian pyramid (left column) and Laplacian pyramid (right column) of a terrain model. Lower levels in both pyramids are enlarged to the size of the top levels.

A Laplacian pyramid is derived from a Gaussian pyramid. A Gaussian pyramid consists of a series of raster grids, which are blurred using a Gaussian filter and scaled down (left column in Figure 1). Blurring and scaling are applied multiple times, creating a stack of successively smaller grids, with each pixel containing a local average that corresponds to a neighborhood of twice its size on a more detailed level of the pyramid. Hence, a series of Gaussian low-pass filters are applied to build a Gaussian pyramid, successively removing high-frequency details. The levels are labeled Gi in the left column of Figure 1.

The Gaussian pyramid is then used to derive the Laplacian pyramid. Each level Li of the Laplacian pyramid consists of the difference between two consecutive levels in the Gaussian pyramid (right column in Figure 1). To compute one level of the Laplacian pyramid, two levels of the Gaussian pyramid are first brought to the same grid size and then corresponding values of the two grids are subtracted. The resulting levels of the Laplacian pyramid encode successive frequency bands. For example, the level L5 in Figure 1 contains the terrain's highest frequency range.

The original terrain model can be reconstructed from the Laplacian pyramid by summing all its levels, and adding the base level of the Gaussian pyramid. Using the labels of Figure 1, this can be expressed by:

$$A = G_1 + (G_2 - G_1) + (G_3 - G_2) + \ldots + (G_6 - G_5) = G_1 + \sum L_i \quad (1)$$

Where:
A = reconstructed grid
G_i = levels of the Gaussian pyramid
L_i = levels of the Laplacian pyramid

The reconstructed grid A is identical to the original unfiltered grid (G_6 in Figure 1). Equation 1 implicitly assigns a unary weight to each level of the Laplacian pyramid. To selectively amplify or attenuate frequency bands, a weighted reconstruction is needed:

$$A' = G_1 + \sum w_i L_i \quad (2)$$

Where:
A' = reconstructed grid
G_1 = smallest level of the Gaussian pyramid
L_i = levels of the Laplacian pyramid
w_i = weight coefficients for L_i

The user interactively adjusts the weight coefficients wi. Note that the smallest level of the Gaussian pyramid G_1 in Equation 2 could also be scaled by an additional weight coefficient. But since this weight is not of much practical use, Equation 2 and all following equations are simplified and do not contain this weight coefficient.

Figure 2 shows the visual effect when attenuating high frequency bands and exaggerating a mid-frequency band. The weight coefficients in this

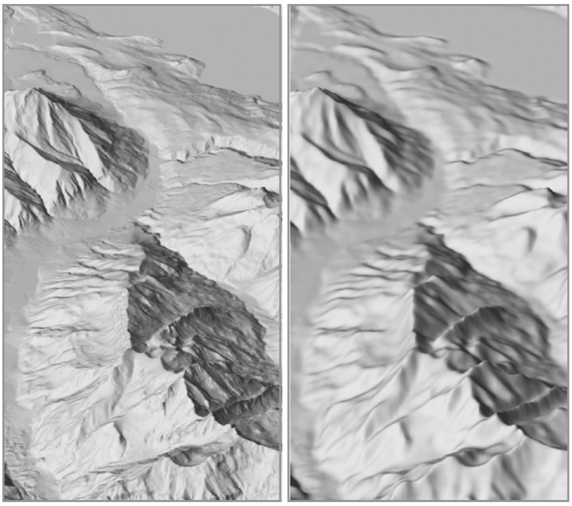


Figure 2. Removing high-frequency details and amplifying a mid-frequency band. Left: original terrain; right: filtered terrain.

particular example are: w_1 = 100%, w_2 = 100%, w_3 = 200%, w_4 = 25%, and w_5 = 0%. The amplification w_3 of the mid-frequency band L_3 accentuates valleys, while the highest frequency band is completely removed by setting w_5 to 0.

Curvature for Edge Preservation

Filtering high-frequency bands removes small details, but also overly smoothes characteristic sharp-edged break lines, such as ridge and valley lines. The basic idea is to automatically identify topographic break lines and apply different weight coefficients to their neighborhood. Hence, the basic model of low-pass filtering is improved by including a second operation to preserve discontinuities and thus restrict simplification to the regions between them. The proposed method modifies the traditional reconstruction of Laplacian pyramids. The traditional reconstruction as described in the previous section and expressed in Equation 2 applies isotropic weight coefficients to the frequency bands, i.e., coefficients do not change with location. In our approach for terrain generalization, the coefficient for each frequency band changes spatially, i.e., anisotropic weight coefficients replace the isotropic ones in Equation 2:

$$A' = G_1 + \sum w_{xyi} L_i \qquad (3)$$

Where:

w_{xyi} = anisotropic weight coefficients at position x and y for L_i

A multitude of methods exist for identifying topographic break lines (Peucker and Douglas 1975; Seemuller 1989; Takahashi et al. 1995). We use curvature coefficients for locally varying the weights. Curvature coefficients are appropriate for modeling geomorphometric elements like ridges and valleys (Wood 1996), and have been applied for the generalization of terrain for relief shading (Leonowicz et al. 2010). We use the plan curvature index, which measures the rate of change of aspect along a contour line in the horizontal plane; it differentiates between convex and concave forms, and defines sharp and clear lines of ridges and valleys (Wilson and Gallant 2000). The anisotropic weight coefficients wxyi are computed as follows to take curvature into account when reconstructing the terrain from the Laplacian pyramid:

$$w_{xyi} = u_i + v_i\, c_{xyi} \qquad (4)$$

Where

u_i = user defined global weight parameter for L_i

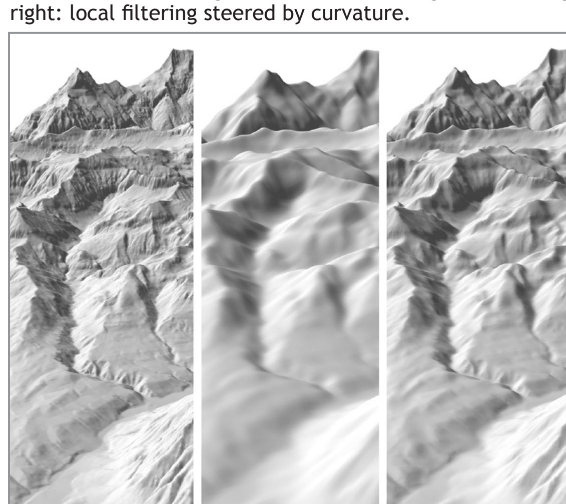
v_i = user defined local weight parameter for L_i

c_{xyi} = local plan curvature at position x and y of $G_1 + \sum w_{xyi-1} L_{i-1}$

u_i and v_i are weight parameters defined by the user. The user adjusts global or isotropic filtering with u_i, and anisotropic preservation of ridge lines identified by high curvature values with v_i. The plan curvature c_{xyi} varies with location, and is computed from the sum of frequency bands with frequencies lower than the current band L_i. This approach has the desired effect of accentuating ridge lines: At locations where lower frequencies contain concave or convex shapes, terrain curvature is further accentuated with higher-frequency details of the added band L_i (Figure 3).

Plan curvature is calculated with the Evans–Young method, which uses a quadratic trend surface for each cell, fitted to the 3×3 local elevation matrix. For formulas see Pennock et al. (1987), Wood, (1996) and Shary et al. (2002). Two transformations are applied to the Evans-Young coefficient to find c_{xyi} in Equation 4. First, coefficients are computed for all terrain cells, then stored in a grid and blurred by convolving with a two-dimensional Gaussian bell curve. Without this blurring, spiky structures would appear along break lines in the reconstructed terrain. The second transformation is an exponentiation applied to the blurred curvature coefficients. An exponent close to

Figure 3. A second set of weight coefficients for preserving break lines. Left: original terrain; middle: global filtering; right: local filtering steered by curvature.



0 results in mountain slopes with many details, while an exponent close to 1 adds high frequency details along break lines. The user can adjust the size of the blurring kernel and the exponent.

The procedure outlined so far concentrates on convex ridge lines and ignores the generalization of concave valley lines. However, a third set of user-defined weight coefficients can be applied along concave lines in a similar way. The parameter v_i in Equation 4 for weighting the influence of the curvature coefficient is either chosen from the user-defined coefficient set for ridges or from the set for valleys – depending on whether the frequency band L_i adds convex or concave details. To determine what type of curvature is added, the local profile curvature index of the frequency band L_i is computed (Wilson and Gallant 2000). Figure 3 illustrates the effect of a second set of weight coefficients for mountain ridges (i.e., convex curvature), and a third set of coefficients for valleys (i.e., concave curvature). The figure in the middle shows attenuated high and mid-frequency bands. The figure on the right restores details along ridge and valley lines by amplifying the coefficients of the frequency bands attenuated before.

View Dependent Filtering for 3D Maps

As outlined in the introduction, the level of terrain generalization must increase with the distance to the viewer of the scene when central perspective projection is used. The proposed method can be extended to variably adjust the level of generalization in the foreground and the background. Two sets of weight coefficients are defined, one for the foreground and one for the background. The user defined weight coefficients u_i and v_i in Equation 4 are then linearly interpolated between the foreground and the background. Equation 5 exemplifies the linear interpolation for the coefficient u_i. The interpolation of the coefficient v_i is identical.

$$u_i = u_d\, u_{ifore} + (1 - u_d)\, u_{iback} \qquad (5)$$

Where:

u_i = the blended weight coefficient
u_{ifore}, u_{iback} = user defined foreground and background coefficients for L_i
u_d = distance-dependent weigh factor: 1 in foreground, 0 in background

Figure 4 shows the type of generalization effect that can be achieved. On the right side of Figure 4, details in the highest frequency band are slightly attenuated in the foreground, and ridges in the highest frequency band are moderately accentuated. In the background, the three highest frequency bands are completely removed, except for details along ridges, which are accentuated to preserve the spiky character of this alpine terrain.

Masked Frequency Filtering

The amplification of mid-frequency bands accentuates ridges and valleys, but also creates disturbing concave depressions along valley bottoms where steep slopes adjoin flat planes. This problem can be alleviated with a mask for flat areas. One approach is to extract slope indices, and limit the amplification of mid-frequency bands to flat terrain. Other types of masks can be used as well, for example, a cartographically designed mask with lakes and valley bottoms. Whatever type of mask used, a simplistic combination of (a) the terrain reconstructed from a weighted Laplacian pyramid with (b) the original terrain would result in discontinuities in the combined terrain. Visually disturbing stepped structures would appear, because the reconstructed terrain is likely to have locally shifted in vertical direction due to user defined weight parameters. Hence, masked areas must be taken into account when reconstructing the terrain from the Laplacian pyramid:

$$w_{xyi} = (u_i + v_i\, c_{xyi} - 1)\, m_{xyi} + 1 \qquad (6)$$

Where:

m_{xyi} = anisotropic mask value at position x and y, [0 … 1]

The mask is first transformed to a Gaussian pyramid, and

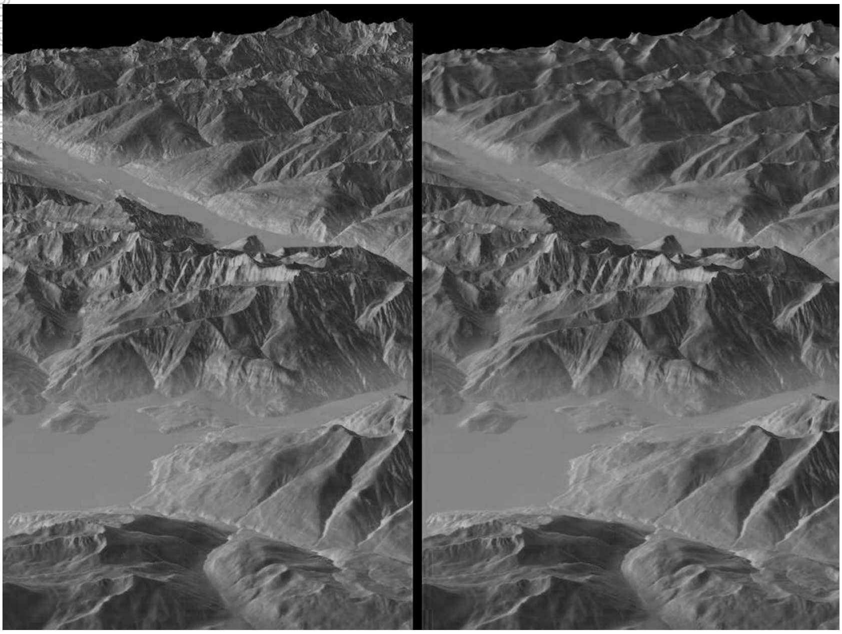


Figure 4. Increasing filtering from foreground to background. Left: original terrain; right: filtered terrain, with ridges accentuated in the background.

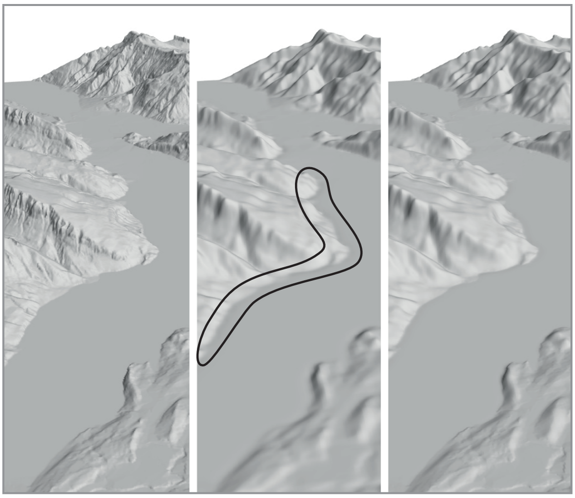


Figure 5. Masks for avoiding artifacts in flat areas. Left: original terrain; middle: globally filtered terrain with depressions highlighted; right: filtering limited to steep areas.

m_{xyi} is the corresponding level i of this pyramid. The mask weight m_{xyi} locally varies between 0 and 1. If m_{xyi} is 0, the frequency band L_i is added with no filtering applied, and locally the unfiltered original terrain model is reconstructed. If m_{xyi} is 1, Equation 6 simplifies to Equation 4. Figure 5 illustrates the masking effect when slope values mask flat areas: artifacts are reduced where the steep slopes adjoin the flat plane.

Implementation and Conclusion

The described method for generalizing digital terrain was implemented in Terrain Equalizer, a free and open-source application available at *http://www.terraincartography.com/*. Terrain Equalizer was written in Java and runs on all major platforms. For rendering previews of the generalized terrain model, Terrain Equalizer builds on JOGL, a Java wrapper for OpenGL (JOGL 2009; OpenGL.org 2008). Terrain Equalizer's interface features sets of vertical sliders for adjusting the weight coefficients of individual frequency bands, using an equalizer metaphor.

As outlined in the section on related work, various alternatives exist to Laplacian pyramids for selectively manipulating frequency bands, such as low-pass filters or wavelet and Fourier transforms. Compared to these alternatives, the method presented here is easy to comprehend and control by the user. Also, the algorithm is relatively simple to implement, lends itself to multithreading, and is fast enough for interactive manipulation of terrains with up to several million cells.

Frequency-based generalization is somewhat limited. Terrain can only be simplified or emphasized, which is also the case for our approach. Other generalization operators, such as displacement or combination operators, are not addressed by our research. Weibel (1992) suggested generalization operators imbedded in an interactive graphical environment, since the objective of cartographic terrain generalization is not so much noise removal as it is graphical simplification. Terrain Equalizer offers such a user interface. We applied it for 3D mapping using terrain models of varying resolution and origin. We were able to successfully remove disturbing high-frequency details, while preserving important break lines; and we successfully enhanced important terrain structures by amplifying mid-frequency bands. Filtering parameters could easily be adjusted to the terrain character and to various levels of generalization. Furthermore, it proved to be easy to adjust the level of generalization to the distance from the viewer. However, we encountered problems where steep slopes and flat valleys meet, when amplifying mid-frequency bands. These problems are at least partially solved by the inclusion of masks – nevertheless, their integration could be improved and easer to control.

Various experts have tested Terrain Equalizer and provided encouraging feedback. A more detailed quantitative analysis or a comparison with other generalization methods is difficult. As with all research in cartographic generalization, only qualitative impressions can be gained from visual analysis to assess the suitability of a generalization operator. Ultimately, only a subjective evaluation of the visual results can tell whether the generalization operation was successful.

Parameters are seamlessly interpolated from the background to the foreground throughout a 3D map. The portability of such a terrain with a generalization level varying from foreground to background is obviously limited, as it can only be viewed in a single predefined direction, once the filtering is applied. This limitation could be overcome for interactive terrain visualizations – where the user can freely choose the position and direction of view – by generalizing the terrain in real time. Interactive frame rates could possibly be achieved by porting the Laplacian pyramid to modern GPUs, as proposed by Strengert et al. (2006) for image processing.

Terrain Equalizer and the underlying algorithms were developed for generalizing digital terrain of mountainous areas at large and medium scales, specifically for the representation as 3D maps. Alternative applications of Laplacian pyramids constrained by curvature remain to be explored for terrain generalization and visualization. Particularly, this approach could be used to remove artifacts from SRTM or LIDAR terrain data, and it could also be applied to generalize terrain before generating 2D visualizations, such as contour lines or shaded relief.

ACKNOWLEDGEMENTS

The authors would like to thank the anonymous reviewers for their valuable comments.

REFERENCES

Amgaa, T. 2003. *Wavelet-based analysis for object separation from laser altimetry data*. Master's thesis, ITC, Enschede, NL.

Bjorke, J. and S. Nilsen. 2003. Wavelets applied to simplification of digital terrain models. *International Journal of Geographical Information Science* 17(7): 601–621.

Burt, P. J. 1981. Fast filter transform for image processing. *Computer Graphics and Image Processing* 16(1): 20–51.

Burt, P. J. and E.H. Adelson. 1983a. The Laplacian pyramid as a compact image code. *IEEE Transactions on Communications* 31(4): 532–540.

Burt, P.J. and E.H. Adelson. 1983b. A multiresolution spline with application to image mosaics. *ACM Transactions on Graphics* 2(4): 217–236.

Danovaro, E., L. De Floriani, L. Papaleo and M. Vitaliet. 2006. A multi-resolution representation for terrain morphology. In: M. Raubal et al. (eds.) *GIScience 2006, Lecture Notes in Computer Science* 4197: 33–46.

Desbrun, M., M. Meyer, M. Schröder and A.H. Barr. 2000. Anisotropic feature- preserving denoising of height fields and bivariate data.In: *Proceedings of the Graphics Interface 2000 Conference*, pp. 145–152.

Fei, L. and J. He. 2009. A three-dimensional Douglas-Peucker algorithm and its application to automated generalization of DEMs. *International Journal of Geographical Information Science* 23(6): 703–718.

Gallant, J. C. and M.F. Hutchinson. 1996. Towards an understanding of landscape scale and structure. *Third International Conference/Workshop on Integrating GIS and Environmental Modeling*, Santa Fe, NM, USA.

Hofer, M., G. Sapiro and J. Wallner. 2006. Fair polyline networks for constrained smoothing of digital terrain elevation data. *IEEE Transactions on Geoscience and Remote Sensing* 44(10): 2983–2990.

JOGL, 2009. *Java binding for the OpenGL API Wiki*, http://kenai.com/projects/jogl/pages/Home (accessed 13 September 2010).

Jordan, G. 2007. Adaptive smoothing of valleys in DEMs using TIN interpolation from ridge elevations: an application to morphotectonic aspect analysis. *Computers & Geosciences* 33(4): 573–585.

Kalbermatten, M. 2010. *Multiscale analysis of high resolution digital elevation models using the wavelet transform*. PhD thesis, EPFL Lausanne.

Leonowicz, A.M., B. Jenny and L. Hurni. 2010. Automated reduction of visual complexity in small-scale relief shading. *Cartographica* 45(1): 64–74.

Li, Z. 2008. Multi-scale digital terrain modelling and analysis. Advances in Digital Terrain Analysis, *Lecture Notes in Geoinformation and Cartography*, Springer Berlin Heidelberg: 59–83.

Mahler, E. 2001. *Scale-dependent filtering of high resolution digital terrain models in the wavelet domain*. Master's thesis, Department of Geography, University of Zurich.

Mohan, A., A. Bartesaghi and G. Sapiro. 2006. Constrained regularization of digital terrain elevation data. *IEEE Geoscience and Remote Sensing Letters* 3(1): 59–62.

OpenGL.org, 2008. OpenGL – the industry's foundation for high performance graphics, http://www.opengl.org (accessed 13 September 2010).

Pajarola, R. and E. Gobbetti. 2007. Survey on semi-regular multiresolution models for interactive terrain rendering. *The Visual Computer* 23(8): 583–605.

Pennock, D. J., B.J. Zebarth and E. de Jong. 1987. Landform classification and soil distribution in hummocky terrain, Saskatchewan, Canada. *Geoderma* 40: 297–315.

Peucker, T. K. and D.H. Douglas. 1975. Detection of surface-specific points by local parallel processing of discrete terrain elevation data. *Computer Graphics and Image Processing* 4(4): 375–387.

Seemuller, W. W. 1989. The extraction of ordered vector drainage networks from elevation data. *Computer Vision, Graphics, and Image Processing* 47(1): 45–48.

Shary, P.A., L.S. Sharaya and A.V. Mitusov. 2002. Fundamental quantitative methods of land surface analysis. *Geoderma* 107: 1–32.

Silván-Cárdenas, J. L. and L. Wang. 2006. A multi-resolution approach for filtering LiDAR altimetry data. *ISPRS Journal of Photogrammetry & Remote Sensing* 61: 11–22.

Strengert M., M. Kraus and T. Ertl. 2006. Pyramid methods in GPU-based image processing. *Proceedings of Vision, Modeling, and Visualization* 2006: 169–176.

Takahashi, S., T. Ikeda, Y. Shinagawa, T.L. Kunii and M. Ueda. 1995. Algorithms for extracting correct critical points and constructing topological graphs from discrete geographical elevation data. *Computer Graphics Forum* 14(3): 181–192.

Tasdizen, T. and R. Whitaker. 2003. Feature preserving variational smoothing of terrain data. *Proceedings of the 2nd International IEEE Workshop on Variational, Geometric and Level Set Methods in Computer Vision*.

Weibel, R. 1992. Models and experiments for adaptive computer-assisted terrain generalization. *Cartography and Geographic Information Science* 19(3): 133–153.

Weibel, R. and M. Heller. 1991. Digital terrain modelling. *Geographical Information Systems: Principles and Applications*: 269–297, Longman, London.

Wilson, J.P. and J.C. Gallant. 2000. *Terrain Analysis: Principles and Applications*. New York: Wiley.

Wood, J.D. 1996. *The geomorphological characterisation of digital elevation models*. PhD thesis, University of Leicester.