Grazer Schriften der Geographie und Raumforschung **Band 43 / 2007** pp. 235 - 242

A New Semi-Automatic Tool for 3-Dimensional Landform Mapping

Martin Schneider (1) and Jan-Christoph Otto (2)
(1) Computer Graphics Group, University Bonn, Germany
(2) Department of Geography, University Bonn, Germany

Abstract

Landform mapping on digital aerial photos and digital elevation models (DEM) using GIS is commonly performed manually on a plane two-dimensional data set. Several derivatives of the base data (slope-, aspect-, curvature maps, image classification) can be used to enhance the identification of landforms on a two-dimensional image. Our natural three-dimensional (3d) perception of the world is disabled by the restriction of digitisation tools using 2d visualisation techniques. We introduce two new semi-automatic tools for 3d landform mapping. The tool is integrated into a real-time 3d visualisation engine, which allows interactive exploration of a virtual terrain based on aerial photo and high resolution DEM data. The user is enabled to perform the mapping directly in this 3d virtual terrain. In addition to the 3d capabilities the tool assists the user by specifying geomorphological objects quickly and accurately by using simple mouse gestures. This is realised by the utilisation of digital image segmentation techniques applied to the image data. In contrast to tedious and time consuming manual segmentation user input is minimised without sacrificing accuracy.

We mapped meso-scale geomorphologic landforms in a high alpine valley using both traditional 2d GIS methods (ArcMap, ESRI) and the new semi-automatic tool. A digital high resolution data set (HRSC (High Resolution Stereo Camera), DEM, 1m, and aerial photos) served as a database for the 3d visualisation engine as well as the GIS. We compared the different methods here with respect to applicability, accuracy and mapping efficiency.

KEY WORDS: Geomorphological mapping, boundary detection, interactive segmentation, Intelligent Scissors, Graph Cut, 3d visualisation, HRSC data

1. Introduction

Landform mapping is often the basis for various kinds of geomorphological investigations. The resulting geomorphological map is the standard tool to perceive and investigate an area at focus in a complex and holistic way. Geomorphological maps compile knowledge on landforms, surface processes and surface materials that have a widespread application in land management practices, natural hazard assessments or landform evolution studies (Cooke and Doornkamp 1974; Otto and Dikau 2004; Seijmonsbergen and de Graaff 2006). Traditionally, geomorphological mapping is based on field work supplemented by the interpretation of aerial photographs and literature research. The availability and usage of high resolution digital elevation models, satellite and aerial images enables to avoid or reduce the amount of the time-consuming and costly field work, especially in remote and highly dynamic regions like high mountains. Modern GIS tools facilitate the compilation, production and distribution of geomorphological maps. Enhanced digitising tools, data layers, data base functions, symbol creation, print and web publishing are some of the enhancements provided by GIS. However, most of the cartographic features of GIS software are limited to a two-dimensional working environment.

Geomorphological mapping decomposes land surface complexity into structural patterns, landforms and landform elements. Recent developments in remote sensing techniques and GIS aim at an automated recognition of geomorphological objects on digital terrain data. Automatic landform recognition is performed using elevation data only (Schmidt and Hewitt 2004; van Asselen and Seijmonsbergen 2006), or by combining elevation data and imagery information (Schneevoigt and Schrott 2006). However so far, automatic recognition suffers from land surface complexity and its continuum character, represented by vague landform boundaries, overlapping landforms and a great variety of structural properties. Consequently, automatic landform recognition is restricted to landform elements or units, while single landforms cannot be identified in detail. A detailed geomorphological map still requires the manual landform digitalisation, whether transferred from previously acquired field data, or genuinely digitised from remote sensing data on screen. The accuracy of digital landform mapping depends on the resolution of the terrain data, the visual perception of the virtual land surface morphology and the diligence and knowledge of user. However, a simple, two-dimensional bird's eye view visualisation of imagery and elevation data restricts the perception of landforms to strong contrasts and large objects. Derivatives of elevation can be used to accentuate morphology changes and break lines in the land surface (Smith and Clark 2005). For example relief-shading is commonly used for visualisations; however, it is prone to biasing due to the variable azimuth of the light source. The increasing availability of digital elevation data more and more replaces the conventional stereoscopic aerial photo interpretation by combining image and elevation data in a three-dimensional (3d) visualisation environment and simulates the natural 3d perception on the screen. However, up to now, 3d visualisation software is usually restricted to a simple data inspection without the possibility to extract information from the data. We present here two semi-automated tools that enable digitising of objects within a 3d visualisation environment. A visualisation in 3d displays the landform structure in its natural appearance similar to the perception in the field. Moreover, by navigating in the 3d environment landforms can be inspected from arbitrary views including perspectives hardly possible in nature. Landform mapping is performed here using high resolution HRSC (High Resolution Stereo Camera) image and elevation data with a geometrical resolution of 1 m. The HRSC data was produced for a meso-scale alpine valley in the southern Swiss Alps (Turtmann valley, Valais, Switzerland).

2. Methods

Landform mapping extracts single objects from the land surface continuum, defining sharp boundaries between parts of the land surface even if some geomorphological objects may not have sharp limits. The task in object extraction is to specify which parts of the surface belong to an object and which do not. Since the computer is still a long way from doing this work completely unassisted, the user has to define each object individually with pixel accuracy. The tediousness of this pixel-exact work makes object extraction a particularly time-consuming and frustrating task for users. The challenge is therefore to develop a way to specify an object that is less tedious and faster than marking every pixel individually but without sacrificing quality.

There are two main approaches that improve on standard pixel-level segmentation tools: Boundary-based and region-based methods. Each of these methods takes low-level features of the image (e.g. colour differences) and uses them to help automate or guide the object extraction process. Boundary-based methods specify an object by enabling the user to track its boundary with an evolving curve. The user traces along the object boundary and the system optimises the curve in a piecewise manner. Region-based methods work by allowing the user to give vague hints as to which parts of the image are occupied by the object. These hints usually take the form of clicking or dragging on different parts of the image, and are thus quick and easy to do. An underlying optimisation algorithm extracts the actual object boundary based on the user input hints.

Since it depends on the actual kind of object being extracted which method is superior, we present a boundary- and a region-based tool. The boundary-based method is based on Intelligent Scissors (Mortensen and Barrett 1995) and the region-based method uses Graph Cut optimisation (Boykov and Jolly 2001). Both methods are originally interactive segmentation tools to extract objects from images or photos primarily intended for image composition. We enhance and adapt these methods to make them applicable to mapping geomorphological objects in large terrain data sets in a 3d visualisation environment. The tools are integrated into the rendering system presented in Wahl et al. (2004) which uses a quadtree data structure for an efficient real-time visualisation of digital terrain data. The segmentation tools operate on the same quadtree data structure used for rendering which is advantageous in several respects: data has to be held in memory only once, the quadtree structure allows fast access to subparts of the terrain and different levels of detail are already available. Both methods formalise an image as a graph G = (V, E), where V is the set of all nodes and E is the set of all arcs connecting adjacent nodes. The nodes correspond to the pixels in the image and the arcs represent adjacency relationships with four or eight connections between neighbouring pixels.

Our mapping procedure can be outlined as follows: the user performs the object marking operations directly in the 3d environment. Then, hidden from the user, the input is projected from the terrain surface to the underlying aerial photo on which the actual object extraction is performed. The resulting boundary is projected back on the 3d terrain surface giving immediate feedback to the user and providing the impression to work directly in 3d. The extracted objects can be exported in the common shapefile format for further processing in standard GIS.

2.1. Intelligent Scissors

Intelligent Scissors formulates the boundary detection problem in an image as an optimal path search in a graph. The objective is to find the optimal path from a seed node to a destination node where pixels in the image represent nodes and edges exist between neighbouring pixels (Fig. 1). An optimal path is defined by the minimum cost path, i.e. a path with the smallest sum of edge costs. Since an optimal path in the graph should correspond to an object boundary in the image, pixels with strong edge features in the image should to lead to low local costs in the graph and vice-versa. To this end, local costs are created as a weighted sum of the following edge features: Laplacian zero-crossing fz, gradient magnitude fg and gradient direction fd. Letting L(p,q) represent the local cost on the directed edge from node p to a neighbouring node q, the local cost function is:

$$L(p,q) = wz \cdot fz(q) + wg \cdot fg(q) + wd \cdot fd(p,q),$$

where each w is an empirically chosen weight of the corresponding feature function (see Mortensen and Barrett 1995 for details).

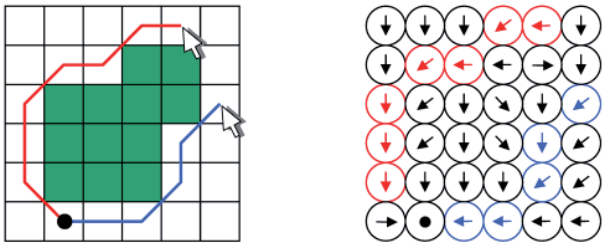

Figure 1

The user starts the segmentation process by planting an initial seed point by simply clicking with the mouse. A path map is constructed that provides the minimum cost path from the seed node to every other node. By moving the mouse, a path from the seed point to the current cursor position is interactively displayed snapping to nearby object boundaries. If the path deviates from the desired object boundary, a new seed point can be established by the user fixing the current path segment as an identified boundary segment and starting a new boundary segment to be extended from the new seed. Placing seed points directly on an object's boundary is often difficult and results in spikes if the seed is not localised exactly. To facilitate seed point placement image snapping (Gleicher 1995) is supported which automatically moves the mouse pointer to nearby image features within a user specified neighbourhood.

The main problem that arises when applying the algorithm to terrain data is the sheer size of the aerial photos often comprising several gigabytes of data. Since the original Intelligent Scissors algorithm is not able to compute the image features and the path map in reasonable time for images of this size, not to speak of memory requirements, we modified the original algorithm in several respects to ensure interactive feedback even on large aerial photos. Computational costs and memory requirements are drastically reduced by exploiting the quadtree representation in two ways. Within a quadtree level we utilise the quadtree tiling to localise the search domain similar to the approach by Kang and Shin (2002). Instead of applying the algorithm on the complete image we only run it on the quadtree tiles containing and surrounding the actual mouse position. As the mouse pointer is moved by the user the search domain is incrementally extended and the path map is updated accordingly. For reasonable user input, i.e. when the mouse is located sufficiently close to the desired object boundary, there is no difference to the original method.

Secondly, we exploit the hierarchical structure of the

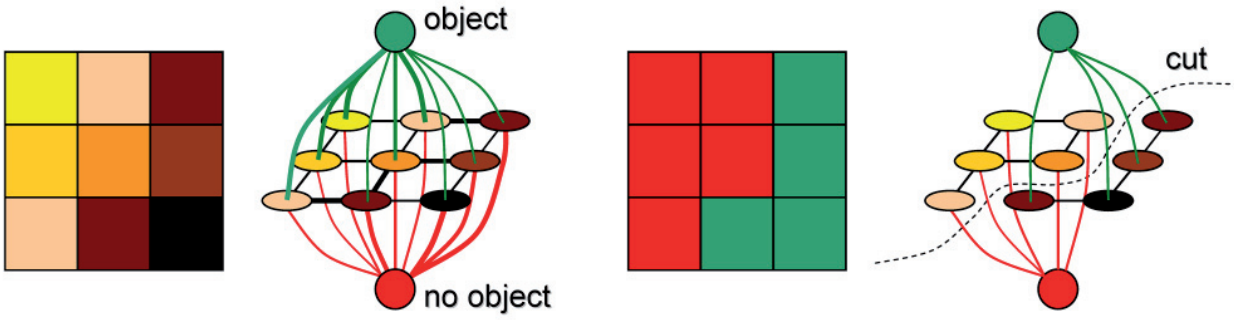

Figure 2

quadtree by starting segmentation not on the finest quadtree level but on a coarser one. The solution computed on the coarse level is then propagated to the next higher resolution level. Here we perform the segmentation only within a narrow band surrounding the projected result from the coarser level. This procedure is repeated until the highest resolution level is reached. Since the algorithm is only run on the subgraph that comprises the narrow band, the additional computation at the fine resolution is significantly less than running it on the full graph. Additionally, since only the subgraph has to be established memory consumption is also drastically reduced.

2.2. Graph Cut

The region-based object extraction algorithm presented here is based on works by Li et al. (2004). Object extraction is formulated as a binary labelling problem which assigns to each node p, i.e. pixel, a unique label xp {object present, object absent} (Fig. 2). The solution X = {xp} can be obtained by minimising a Gibbs energy (Geman and Geman 1984):

$$G(X) = \sum R(x_p) + y\sum B(x_p, x_q),$$

where R(xp) is the likelihood energy encoding the cost when the label of node p is xp, i.e. it encodes the likelihood that the pixel is part of the object. The likelihood is estimated as the colour similarity of the pixel's colour to the colours of the pixel assumed to belong to the object or the surrounding respectively. The colour similarity is actually computed by clustering the colours of each of the two regions using K-Means (Duda et al. 2000) (alternatively, other techniques can be used here to model the colour distribution, e.g. Gaussian mixture models as in Rother et al. (2004). The prior energy B(xp, xq) denotes the cost when the labels of adjacent nodes p and q are xp and xq respectively. It represents the energy due to the gradient along the object boundary. B is defined as a function of the colour gradient between two nodes p and q. In other words, B is a penalty term when adjacent nodes are assigned with different labels. The more similar the colours of the two nodes are, the larger B is, and thus the less likely the edge is on the object boundary. The influence of the region and boundary terms is controlled by the weighting factor γ. Decreasing γ leads to more complex and longer boundaries and generally increases the number of resulting objects, especially small ones. The energy function G(X) is minimised using the max-flow algorithm presented in Boykov and Kolmogorov (2001) which is specially designed for vision problems.

To specify an object, the user marks a few lines by dragging the mouse cursor while holding a button (left button indicating presence, right button absence of the object). A green or a red line is displayed for indicating the object or the surrounding respectively. This high-level painting user interface does not require very precise user inputs since usually marking lines need not be very close to the actual boundary. First of all, user input provides hard constraints for the segmentation. In addition marked pixels are used to initialise the models for the colour distribution. As input for the Graph Cut optimisation we consider only the part of the image inside the bounding rectangle of the user input. The quality of the results can be further improved by iterating the procedure (Rother et al. 2004), alternating between estimation (Graph Cut optimisation) and parameter learning (updating the colour distribution model). It also turned out to be useful to increase γ in each iteration step. The idea behind this is to allow the generation of a reasonable colour distribution in the early iteration steps, while putting emphasis on finding a good boundary in later iterations. If, however, the result is not satisfying, further user editing is necessary. Roughly marking additional lines in the incorrectly labelled areas can be used to update the result appropriately.

Despite the fact that the Graph Cut technique provides a powerful tool for image segmentation, the time and memory consumption constrain its feasibility in many applications where large data sets need to be processed. Since it is crucial that our system generates the object boundary with very little delay, we apply the multilevel banded heuristic presented in Lombaert et al. (2005) to drastically reduce both running time and memory consumption of the Graph Cut optimisation. The procedure

is similar to the heuristic we used to accelerate Intelligent Scissors described in the previous chapter. First the Graph Cut is solved on a coarse level graph. Then the Graph Cut is solved at successive higher resolution levels but only on a narrow banded graph derived from the solution found at the previous coarser graph.

3. Study area

We used a HRSC data set of the Turtmann valley in Switzerland. The app. 110 km² large alpine catchment, located in the southern mountain range of the Pennine Alps, contains a full set of high Alpine geomorphological landforms that have been mapped using the two tools. The valley is 20 km long and the main glacial trough is oriented from south to north with 15 hanging valleys to the west and east. Approximately 14 % of the valley surface is covered by glaciers. More than 80 recent and relict rock glaciers characterise the hanging valleys conditioned by the dry, continental climatic conditions of the Pennine Alps. Lithologies include palaeozoic schists and gneisses and mesozoic dolomites, limestones, and marbles of the penninic nappe Siviez-Mischabel.

4. Mapping alpine landforms in the Turtmann Valley

For the Turtmann Valley a detailed geomorphological map and a GIS database of landform polygons exists (Otto and Dikau 2004; Otto in prep.) that are based on field work data and manual digitising on the HRSC data using ArcGIS. In order to asses the applicability of the new object extraction tools, different landform types have been mapped again in the 3d visualisation environment and compared to the previously, manually mapped objects.

In evaluating the performance of boundary construction methods, three factors are considered to be of prime

importance: speed, repeatability and accuracy. Accuracy measures the degree of agreement of the extracted boundary with the actual boundary. In general, the use of better cost functions in the algorithms could increase accuracy in tracking strong edge features. Unfortunately, it is hard to define the actual boundary itself since it is not always composed of the edges with the strongest features. Thus, a manually traced boundary is often considered to be a good approximation as the target boundary and is therefore used as ground truth. However, the credibility of manual tracing is still questionable since its repeatability is usually worse than that of (semi)automatic methods. What is more, manual tracing is usually never performed with pixel accuracy resulting in a simplified version of the actual boundary and is therefore too coarse to serve as ground truth when evaluating accuracy of pixel exact methods. We therefore present a qualitative (i.e., visual) comparison of the object boundaries defined with the different tools combined with the number of nodes in the resulting boundary as an indicator of accuracy. We use the amount of user effort in terms of time and mouse clicks required to extract an object boundary as an indicator for digitising speed, i.e. the number of nodes placed in ArcGIS, the number of seed points planted with Intelligent Scissors and the number of mouse strokes used for Graph Cut. Repeatability is inversely proportional to the variation over the boundaries extracted from repetitive experiments.

Fig. 3 depicts a shallow landslide on the western trough slope of the valley caused by the failure of a drainage pipe. The landslide area is digitised manually with 86 nodes (Fig. 3a) in ArcGIS. Using the Intelligent Scissors tool, only about 12 nodes (seed points) need to be placed (Fig. 3b,c). For the Graph Cut method the areas of interest are marked by green lines (Fig. 3d), while area surrounding the object is marked by the red lines. Both tools make full use of the high-resolution of the data set resulting in polygons with more than 1000 nodes that are created within a few seconds. This high number of nodes may be impractical and leads to large file sizes, especially when the number of objects is high as well. As a countermeasure the number of nodes can be reduced using an error-controlled polyline simplification. From visual inspection, the Intelligent Scissors tool delimitates the most precise boundaries of the landslide area compared to the manual mapping. In contrast, the object boundary created by the Graph Cut method appears to be more serrated. Some areas are included within the polygon that have not been mapped manually, for example parts of the gravel road. Other areas, for example the darker parts within the polygon are not considered, although they clearly belong to the landslide area. Thus, some post-editing of the Graph Cut derived object

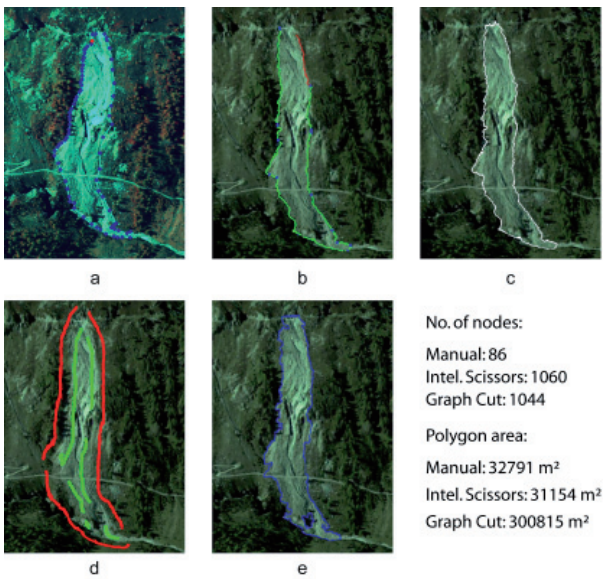

Figure 3

is required to extract a more precise landform boundary. Both methods produce a jagged section in middle part of the left boundary (Fig. 3c, e), where the landslide area crosses the forest. Here, the tools suffer from the lighting situation causing the left boundary to be delimitated by the shadow of the trees. This effect doesn't occur when digitised manually, as the user would consider shadowing effects and create a straighter border.

5. Discussion and perspectives

The mapping of landforms with the two tools proves to be very accurate and fast. Within a few steps, high resolution objects, represented by a high number of nodes and accurate boundaries can be generated within a short time. Additionally, the 3d visualisation environment offers numerous ways to perceive the landform and its boundaries not only on the aerial image, but including the landform surface structure. Due to technical differences each tool has individual advantages. The Intelligent Scissors allow a more supervised digitisation, as the location of the automatic path can be controlled by the placement of intermediate seed points. This is very useful for surfaces with little contrast. From its segmentation principle, the Intelligent Scissors especially suited for extraction of linear features and boundaries, like gullies, debris flow tracks, ridges, steps or moraine and rock glacier boundaries. On the other hand, the Graph Cut tool is less controlled and very efficient for mapping of extensive objects. Both methods allow for efficient post-editing. Objects created by the Intelligent Scissors can be changed by editing arbitrary single nodes after the segmentation is terminated. The Graph Cut method enables a repetition of the segmentation process, where further user input can refine the automatic classification. Another advantage of the Graph Cut method is the possibility to extract more than one object at the same time. For example, several snow patches can be extracted simultaneously with a few strokes.

The focus on image data and colour values of both methods has advantages and disadvantages. The lighting conditions are influenced by the landform structure and may increase the discrimination of objects on a surface. However, boundary identification is influenced to a great extent by different lighting situations. Especially in mountainous areas, where strong gradients cause large shadows, landform segmentation is biased, causing the procedures to follow shadow boundaries instead of true landform edges. One approach to overcome this restriction is to include the geometry data of the landform. For example, we modified the Intelligent Scissors tool in order to follow channels (straight – concave landform elements). First tests produced very promising results. Alternatively, the segmentation tools could be applied using derivatives of elevation data as imagery base that are not necessarily biased by lighting. For example, slope, shading and aspect maps could be used as base data. These options will be included in the digitising environment in the future.

Possible applications of the segmentation tools are numerous. Apart from complex geomorphological landforms, a segmentation of simpler features like houses, roads, lakes or fields with clear boundaries is very easily performed. We consider these tools a valuable assistance for landform digitisation that enhance and accelerate landform mapping. Restrictions observed in full automatic classifications using remote sensing techniques do not apply, as the user is in full control of the segmentation steps.

References

BOYKOV, Y. and JOLLY, M. P., 2001: Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images. Proceedings of ICCV, Vol. 1, 105-112.

BOYKOV, Y. and KOLMOGOROV, V., 2001: An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. In Energy Minimization Methods in Computer Vision and Pattern Recognition, 26(9):1124-1137.

COOKE, R.U. and DOORNKAMP, J.C., 1974: Geomorphology in environmental management. Clarendon Press, Oxford, 352-397.

DUDA, R., HART, O.P.E. and STORK, D.G., 2000: Pattern Classification, Wiley Press, Chichester.

GEMAN, S. and GEMAN, D., 1984: Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. IEEE Transactions on Pattern Analysis and Machine Intelligence, 6:721-741.

GLEICHER, M., 1995: Image snapping, SIGGRAPH '95 Proceedings, 183-190.

KANG, H. W., SHIN, S. Y., 2002: Enhanced lane: interactive image segmentation by incremental path map construction. Graphical Models 64(5): 282-303.

LI, Y., SUN, J., TANG, C.K. and SHUM, H.Y., 2004: Lazy snapping. ACM Transaction on Graphics, 23(3), 303-308.

LOMBAERT, H., SUN, Y., GRADY, L. and XU, C., 2005: A multilevel banded graph cuts method for fast image segmentation. Proceedings of Tenth ICCV.

MORTENSEN, E.N. and BARRETT, W.A., 1995: Intelligent scissors for image composition. In SIGGRAPH '95 Proceedings. pages 191-198.

NILSSON N. J., 1998: Principles of artificial intelligence, Springer, Heidelberg.

OTTO, J.C., in prep. Paraglacial sediment storage quantification in the Turtmann Valley, Swiss Alps. Bonner Geographische Abhandlungen. Department of Geography, Bonn.

OTTO, J.C. and DIKAU, R., 2004: Geomorphologic system analysis of a high mountain valley in the Swiss Alps. Zeitschrift fur Geomorphologie, 48(3): 323-341.

ROTHER, C., KOLMOGOROV, V. and BLAKE, A., 2004: "GrabCut" — Interactive foreground extraction using iterated graph cuts, ACM Transactions on Graphics. 23(3).

SCHMIDT, J. and HEWITT, A., 2004: Fuzzy land element classification from DTMs based on geometry and terrain position. Geoderma, 121(3-4): 243-256.

SCHNEEVOIGT, N.J. and SCHROTT, L., 2006: Remote sensing of Alpine landforms. A geomorphic systems approach exemplified in the Bavarian Alps (Reintal, Germany). Geographica Helvetica, 3: in press.

SEIJMONSBERGEN, A.C. and DE GRAAFF, L.W.S., 2006: Geomorphological mapping and geophysical profiling for the evaluation of natural hazards in an alpine catchment. Natural Hazards and Earth System Science, 6(2): 185-193.

VAN ASSELEN, S. and SEIJMONSBERGEN, A.C., 2006: Expert-driven semi-automated geomorphological mapping for a mountainous area using a laser DTM. Geomorphology, 78(3-4): 309-320.

WAHL R., MASSING, M., DEGENER, P., GUTHE, M. and KLEIN, R., 2004: Scalable compression of textured terrain data, Journal of WSCG.

Correspondence to:

Martin Schneider

Research Training Group 437

Computer Graphics Group

University Bonn

Römerstrasse 164

53117 Bonn

Germany

e-mail: ms@cs.uni-bonn.de

Jan-Christoph Otto

Research Training Group 437

Department of Geography

University Bonn

Meckenheimer Allee 166

53115 Bonn

Germany

e-mail: j.otto@giub.uni-bonn.de