

Topological Fragmentation of Medical 3D Surface Mesh Models for Multi-Hierarchy Anatomical Classification

Gerald Adam Zwettler and Werner Backfrieder

Abstract—High resolution 3D mesh representations of patient anatomy with appendant functional classification are of high importance in the field of clinical education and therapy planning. Thereby, classification is not always possible directly from patient morphology, thus necessitating tool support. In this work a hierarchical mesh data model for multi-hierarchy anatomical classification is introduced, allowing labeling of a patient model according to various medical taxonomies. The classification regions are thereby specified utilizing a spline representation to be placed and deformed by a medical expert at low effort. Furthermore, application of randomized dilation allows conversion of the specified regions on the surface into fragmented and closed sub-meshes, comprising the entire anatomical structure. As proof of concept, the semi-automated classification method is implemented for VTK library and visualization of the multi-hierarchy anatomical model is validated with OpenGL, successfully extracting sub-meshes of the brain lobes and preparing classification regions according to Brodmann area taxonomy.

Keywords—3D surface mesh, functional classification in medicine, multi-hierarchy classification, levels of detail, spline-based surface selection

I. INTRODUCTION

MEDICAL atlas representations incorporating anatomical classification models are of high importance for clinical education and research on automated classification methods of segmented patient models. Clinical training applications like VoxelMan [1] derived from the visible human project and many other commercial software toolkits thereby offer detailed insight on the human anatomy from both, a morphological and a functional perception but only cover specific anatomical classifications attached to the delivered 3D models. When building customized medical models incorporating specific anatomical classifications from segmented patient data or available 3D mesh representations, software tool support is required to specify the particular anatomical and functional regions. Wherever the functional classification can be derived from the patient morphology visualized with the particular imaging modality, sophisticated automated classification approaches can be utilized for automated processing. Many automated classification approaches have been presented in the past for

G. Zwettler is with the Research Group Bio- and Medical Informatics, Department of Research and Development at the University of Applied Sciences Upper Austria, Campus Hagenberg, Softwarepark 11, 4232 Hagenberg, Austria (e-mail: gerald.zwettler@fh-hagenberg.at).

W. Backfrieder is with the Department of Biomedical Informatics at the University of Applied Sciences Upper Austria, School of Informatics, Communications and Media, Softwarepark 11, 4232 Hagenberg, Austria (e-mail: werner.backfrieder@fh-hagenberg.at).

various diagnostic domains. Exemplarily, algorithms for fully automated liver lobe classification based on hemodynamic analysis [2] and fully-automated modeling of gyrus and sulcus folds from brain MRI [3] have been developed.

Nevertheless, the vast majority of anatomical classification cannot be performed in a fully automated way, thus necessitating the medical expert to precisely sketch the anatomical regions at maintainable effort. Common selection approaches like pure picking, stencil functionality or lasso contour definition thereby scantily ensure a sufficient level of processing accuracy in case of topological changes or require too much user interaction. Labeling of anatomical regions on a surface mesh allows for advanced visualization but the specified anatomical sub-parts do not represent a closed mesh that is required for allowing full insight into anatomy. To complement the missing hull parts, common concepts like convex hull [4] or VTK based mesh interweaving [5] are applicable but generally do not lead to good approximation of the morphology. Distance map based approaches [8] incur from overlapping region parts and enclave regions generated along the border sections.

In this work a spline-based contour definition concept is presented, allowing specification of complex shapes at the surface of mesh models by placing a low number of control points. Topographic changes are thereby precisely handled and the vertex distances from the spline borders allow for visualization beyond the available triangle resolution. To complement the classification regions to enclosed surface mesh representations, randomized dilation is utilized [9]. Finally, a generic data model for topological fragmentation of 3D mesh representations also applicable to level-of-detail steering is developed. To facilitate multi-hierarchical anatomical classification, each sub-mesh of the topological hierarchy can be assigned classification regions of an arbitrary number of anatomical classifications.

II. METHODOLOGY

For preparation of multi-hierarchical classifications on a surface mesh according to anatomical taxonomies, several steps must be processed. Firstly, all anatomical regions of the particular taxonomy must be marked at the surface at high precision in a semi-automated way. Based on these regions defined on the surface, closed mesh representations can be derived utilizing randomized dilation, allowing further sub-classification. Based on these classification regions placed on the surface and generated closed sub-meshes, the entire input

dataset can be partitioned with respect to resolution as *level-of-detail* (LOD) but also according to anatomy at different levels of granularity. For each of the topologically fragmented meshes, further anatomical classification information can be marked on the surface. The mesh hierarchy as well as the prepared anatomical classifications is modeled in a generic data structure for 3D meshes, developed in the course of this work.

A. Fully-automated Region Labeling at Surface Meshes

If the functional classification scheme is manifested in the anatomical morphology, automated extraction of the target surface areas can be performed. Thus, for liver parenchyma, graph-based analysis on hemodynamics allows for preparation of classification regions according to *Couinaud* taxonomy [2]. In the case of analyzing the gray and white brain matter, topological analysis of the folds on the brain surface allows for determination of gyri and sulci [3]. Nevertheless, if the functional classification is not manifested in anatomical morphology and thus not accentuated by the particular imaging modality, an alternative approach is required. Anatomical classifications are often prepared as medical atlases that are adjusted to the specific anatomical surface models via elastic registration transform. If anatomical taxonomy is thereby represented as texture, the classification regions can be easily determined by thresholding and binarization of the texture map, see Fig. 1.

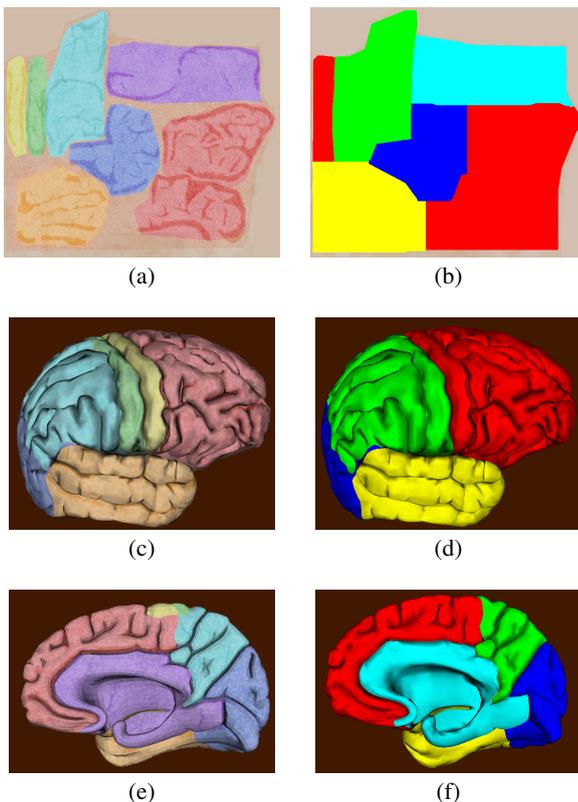


Fig. 1. Classification Regions derived from texture map via simple color segmentation. The brain surface model Human Brain from TurboSquid company is thereby resampled and classified according to frontal, parietal occipital and temporal lobe as well as the cingulate cortex (c-f), utilizing a threshold-based segmentation of the available 2D texture map (a-b).

In this work the fragmentation of the left and right hemisphere into *frontal*, *parietal*, *occipital*, *temporal* lobe and the *cingulate cortex* is achieved via texture map analysis to directly derive the polygons belonging to the particular anatomical regions. In the case of processing anatomical models with no corresponding texture map, a semi-automated concept is required to precisely select the classification regions on the surface mesh.

B. Semi-automated Region Labeling at Surface Meshes

Whenever anatomical structures are represented as polyhedral surface mesh representations, polygons belonging to classification regions can be marked to provide additional input for analysis or visualization utilizing 3D rendering functionality like OpenGL [6] or the *Visualization Toolkit* (VTK) [5]. For selection of regions on the surface, picking represents the core functionality. Picking refers to the selection of vertices, faces and intersection points in 3D world coordinate system via 2D (x, y) coordinate locations placed on the display. The missing z -coordinate is complemented utilizing a depth-buffer readout or by intersection of the projective ray with all mesh polygons, evaluating for a minimum distance to the camera location. To allow for precise selection of polygon groups at low effort, several strategies are applicable:

- depending on the spatial resolution of the input surface mesh, the polygons for the classification regions can be solely selected utilizing face picking. In the case of a huge number of polygons the demand for interaction becomes excessively high and thus this concept is rather impractical.
- stencil functionality allows for selection of polygon groups around a picked marker position, utilizing a predefined radius parametrization. This way the demand for user interaction can be significantly reduced, even for surface meshes at high resolution. The tradeoff between precision and demand for marker positions can be balanced via stencil radius parametrization. Nevertheless, high variability in triangle size, e.g. in case of decimation pre-processing, leads to reduced choices for the selected faces and edges.
- the definition of the target region shape via implicit mathematical functions offers a high level of precision at low demand for user interaction. Thereby, spline representations are utilized to reduce the number of explicitly picked faces to a minimum. The discrete number of spline control points placed at the surface mesh defines the expected region borders as polyline courses. The control points can be dynamically adapted (*move/add/delete*) to achieve the best shape approximation. Nevertheless, there is still an algorithm required to derive the 3D faces enclosed by the spline representation placed at the surface area.

In this work, B-splines are utilized to define the regions of interest at the surface mesh representation. For determination of the polygons enclosed by the spline polyline, two different approaches are feasible:

- a discrete number of equidistant intermediate points can be derived from the polynomial spline representation. By projection of these intermediate points onto the surface, the set of explicit marker positions can be significantly extended. To determine the region borders, the intermediate and explicit marker positions are connected by utilizing *all-pairs shortest path algorithm* (APSP) [7], [10].
- the intermediate points derived from the spline representation are utilized to cut through the surface mesh with the direction derived from the relative camera orientation. Thus, for each vertex of the surface mesh, a test can be performed if it is placed within or beyond the cutting shape borders to determine and thus is added to the classification region or not, respectively.

In course of this research work, both approaches, namely (a) completing the path by utilizing APSP search and (b) cutting through the surface for inside/outside test are implemented and compared. The path completion thereby can produce unexpected results in case of topographic changes, i.e. bumps and sinks relative to the surface. Shortest path search might favor a detour at the same horizontal level in contrast to vertical direction below the spline projected onto the surface mesh, see Fig. 2.

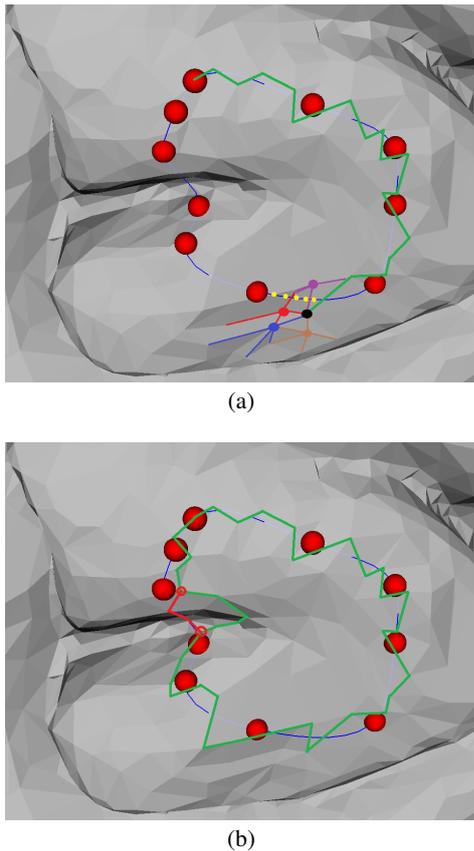


Fig. 2. To connect the intermediate points (blue) between the explicitly picked red spline control points, APSP search is utilized to recursively search for shortest connection to the subsequent control point based on vertex graph (a). In case of topographic changes on the surface, e.g. due to a *sulcus* fold (b), the APSP approach might favor a horizontal detour in contrast to the expected path under the spline projection (red), and thus would more precisely reflect the user-defined spline characteristics.

To better address the problem of topological changes, the cutting approach in Fig. 2 (b) is implemented as precise alternative. According to the camera position when picking the spline marker positions, the course of intermediate points derived from the parametrized splines is interpreted as a cutting pattern, see Fig. 3 (a-b). Thus, for every vertex of the surface mesh, it is evaluated whether the point lies within the cutting area or outside by evaluating the closest distance to the vertical spline borders. The inside / outside test of a particular position is thereby performed by evaluating the normal vectors of the cutting planes in local direction of the closest distance, see Fig. 3 (c).

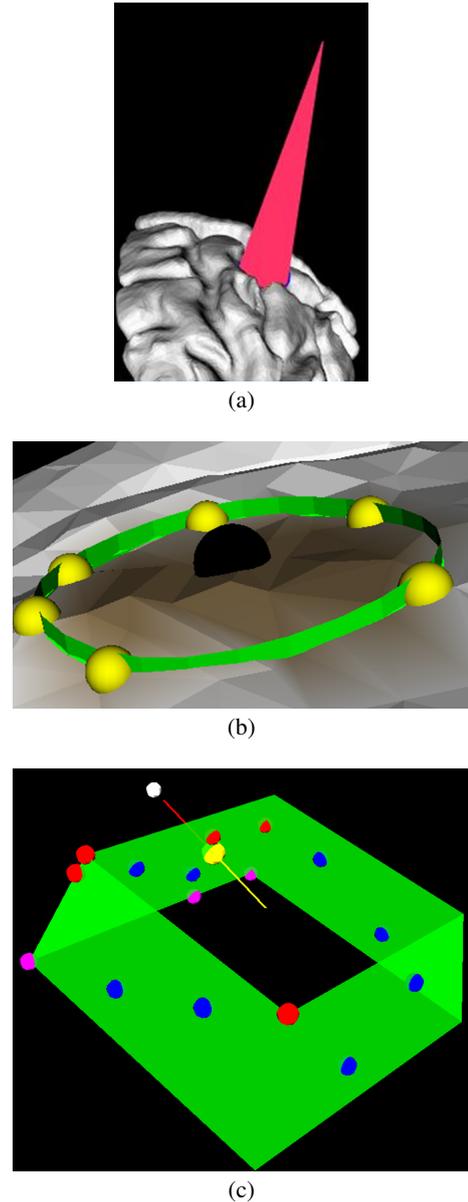
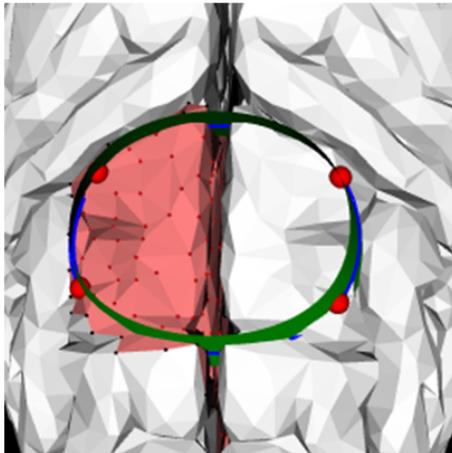


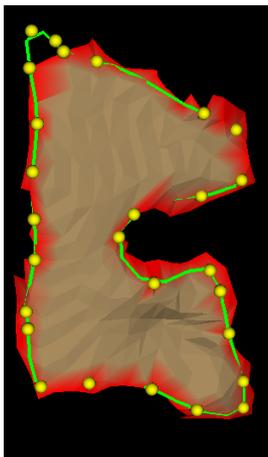
Fig. 3. Spline representation is interpreted as vertical cutting area (b) oriented according to the camera position at time of specifying the control points (a). The inside/outside test (c) is thereby performed by evaluating the closest point (yellow) on the cutting areas for each vertex to evaluate.

As the normal vectors rapidly change in areas with high bending, the sequence of the surrounding $k = 7$ normal vectors is evaluated according to majority voting principle to ensure a

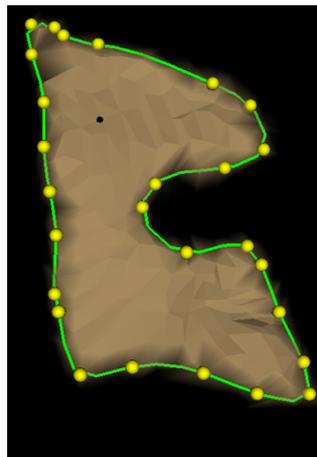
high level of confidence. Besides the boolean result for *inside / outside*, the distance to the borders, thus the confidence of a point belonging to the desired face area, is measured and utilized for advanced visualization, too. Whenever one vertex of a mesh triangle lies within the classification region, the other vertices are complemented in the result set Fig. 4 (a). The distance to the cutting borders is utilized for advanced visualization of faces close to the spline course by utilizing the alpha channel of the color model according to border closeness Fig. 4 (a-b).



(a)



(b)



(c)

Fig. 4. Based on the spline-based region definition (a), the resulting classification areas can follow topographic changes, e.g. in-between the hemispheres. By incorporating the border distance for advanced visualization, the border triangles (red) (b) can be smoothed via alpha transparency (c).

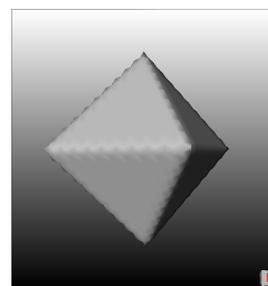
C. Automated Generation of Closed Surface Meshes

As previously discussed, the region classification concepts range from automated analysis of topography or segmentation of attached texture maps to semi-automated contour definition utilizing enclosed splines placed on the surface mesh. Based on the marked areas, enclosed mesh representations are required to feature additional hierarchical fragmentation and to ensure a maximum level of usability. In such a scenario, for each class to discriminate, only the section visible at the surface

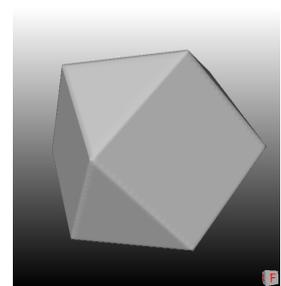
is available. The boarding areas between the neighboring structures inside the surface mesh must be complemented in an automated way. Therefore, Euclidean 3D distance maps per classification structure can be determined to cluster the space inside the surface mesh representation. Nevertheless, the utilization of distance maps is very sensitive to topographic changes on the surface. Furthermore, it is guaranteed that the volume regions assigned to a particular class label are fully connected.

To overcome these limitations, randomized dilation [14] is utilized and evaluated in this work to complete the missing hull parts of the fragmented surface areas. Thereby starting from a voxel representation of the outer hull, iterative dilation is applied to all surface voxels, each time growing the unclassified voxel mass inside the initial surface mesh. Due to incorporated stochasticity of randomized surface processing order, a balancing of the neighboring areas during dilation-driven growth and smooth border areas to be formed in-between are ensured. The resulting voxel representations are re-transformed to 3D mesh data utilizing marching cubes algorithms. To keep the initial mesh quality, the voxel mask dimensionality chosen for intermediate dilation-based classification must be sufficient in size with respect to the particular mean triangle size.

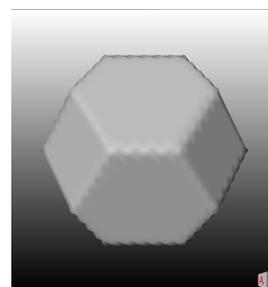
By utilizing N_6 and N_{18} structuring elements in an optimized, alternating manner, growth of a *truncated octahedron* can be approximated in an iterative manner. With a sphericity of $\phi \cong 0.91$, almost spherical growth and thus a good approximation for Euclidean distance based region labeling is achieved. The approach truncated octahedron and the difference compared to 3D sphere representation are presented in Fig. 5.



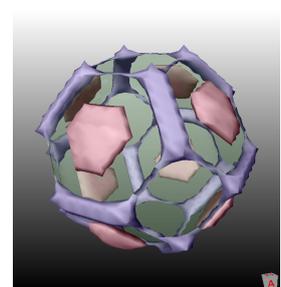
(a)



(b)



(c)



(d)

Fig. 5. Shape approximation with iterative application with structuring elements of size $3 \times 3 \times 3$. While N_6 leads to an approximation of an *octahedron* (b), structuring elements N_{18} approaches shape of a *cuboctahedron*. By optimized alternating utilization of N_6 and N_{18} shape of a *truncated octahedron* (c) can be approximated, showing small differences (d) compared to spherical growth.

III. DESIGN AND IMPLEMENTATION

With respect to the proposed semi-automated classification via splines that are placed at the surface of the derived mesh representations, a full application prototype is developed. The implemented application denoted as *MeshEditor* utilizes VTK [16] for visualization and Java *Eclipse* as IDE, see Fig. 6. To inspect the classified mesh at different hierarchies and levels of resolution, the application framework *MeshViewer*, based on OpenGL and C++ MFC for GUI development, developed by Medical Informatics research group of RISC¹, is extended, see Fig. 7.

To allow for modeling of arbitrary anatomical classifications incorporating mesh hierarchies, a generic 3D data format as delineated in the following code listing is developed in the course of a software-engineering student project. The data structure is thereby presented in *EBNF format* [15] with some syntactical variations. Thereby, sequences are enriched by a scalar value for the expected number of repetitions within brackets where numerical expressions serve as quantifier denoted with prefix #. The meta information regarding the classifications, like textual details concerning the particular anatomical taxonomy or ontology and the color-encoded representation for visualization are specified in an attached *.txt meta info file for matter of flexibility. The developed mesh format allows for the specification of an arbitrary number of medical classifications defined on the surface, utilizing 3D splines for definition of the *Region* instances. Furthermore, a cascading hierarchy of surface meshes can be specified within the dataset, utilizing *Mesh* instances. By specification of an 1 : 1 relationship between parent and child nodes in the hierarchy, equivalent meshes at different granularity are specified in the sense of level-of-detail graduation.

¹<http://www.risc-software.at/de/medical-informatics>, RISC Software GmbH - Medical Informatics, Softwarepark 35, 4232 Hagenberg, Austria

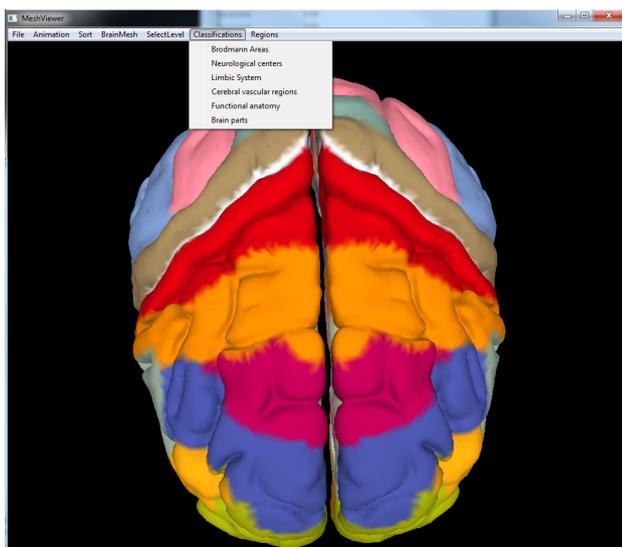


Fig. 7. Hierarchical functional classification integrated into *MeshViewer*. The desired classification taxonomy, hierarchy level and dataset resolution are selected. The spline-based region definition is demonstrated for *Brodmann* areas.

```

1 Header = MetaInfoPath, Mesh;
2 MetaInfoPath = #MetaInfoPathLength, {char}<#
   MetaInfoPathLength>;
3 Mesh = ID_Label, Points, Triangles, Classifications,
   Children;
4 Points = #Points, {Point}<#Points>;
5 Triangles = #Triangles, {Triangle}<#Triangles>;
6 Classifications = #Classifications, { Classification }<#
   Classifications>;
7 Children = #Children {Mesh}<#Children>;
8 Regions = #Regions, {Region}<#Regions>;
9 Indices = #Indices, {Index}<#Indices>;
10 Alphavalues = {Alphavalue}<#Indices>;
11 Classification = ID_Class, Regions;
12 Region = ID_Entry, Indices, Alphavalues;
13 ID_Label = {byte}<2>; //(java, short)
14 ID_Class = {byte}<2>; //(java, short)
15 ID_Entry = {byte}<2>; //(java, short)
16 Point = Guid, Position, NormalVector, TextureCoordinate;
17 Guid = {char}<36>;
18 Position = {double}<3>;
19 NormalVector = {double}<3>;
20 TextureCoordinate = {double}<2>;
21 Alphavalue = {double}<1>;
22 Triangle = {Index}<3>;
23 #MetaInfoPathLength = {byte}<4>; //(java, int)
24 #Points = {byte}<4>; //(java, int)
25 #Triangles = {byte}<4>; //(java, int)
26 #Classifications = {byte}<2>; //(java, short)
27 #Regions = {byte}<2>; //(java, short)
28 #Indices = {byte}<4>; //(java, int)
29 #Children = {byte}<2>; //(java, short)
    
```

IV. RESULTS

The automated closing of the classified surface mesh areas is tested based on brain surface models. For the 4 brain lobes (*frontal, parietal, occipital, temporal*) as well as the *cingulate cortex*, closed surface mesh representations are robustly derived from marked surface areas by utilizing randomized dilation, see Fig. 8 (a-b). The stochastic processing order of the morphological dilation operator thereby ensures smooth border areas and balanced volumes. Compared to distance map based approaches, convex hull generation [4] or the mesh interweaving functionality *vtkFillHolesFilter* provided by VTK, the introduced dilation-based approach best approximates the missing morphological shape.

As proof of concept for the proposed multi-hierarchical classification of topological fragmented 3D surface mesh representations, the classifications Brodmann areas [11] and functional areas are prepared, see Fig. 9. At any node of the topological and level-of-detail sub-mesh hierarchy, an arbitrary number of anatomical classifications can be manifested and utilized by preparing spline-driven classification regions in a semi-automated way.

V. DISCUSSION AND CONCLUSIONS

Methods for semi-automated region classification as well as for the generation of closed sub-meshes based on randomized dilation are presented and evaluated in this work. Together with the developed hierarchical 3D mesh data format, a topological fragmentation of input surface data and the preparation

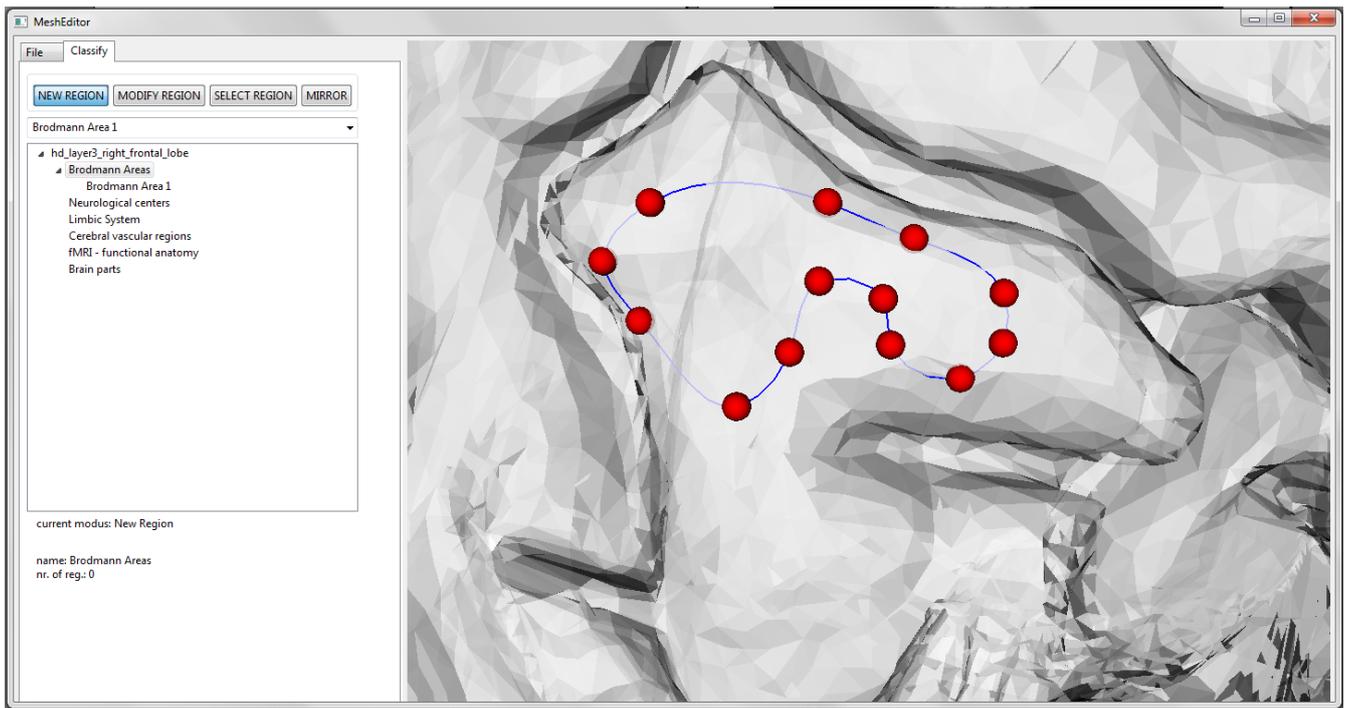


Fig. 6. MeshEditor for spline-based functional classification of surface representations. Defined 3D spline representations allow for specification of various taxonomies, like *Brodman areas*, *neurological centers*, the *limbic system* or the physiological *brain lobes* reflecting morphology.

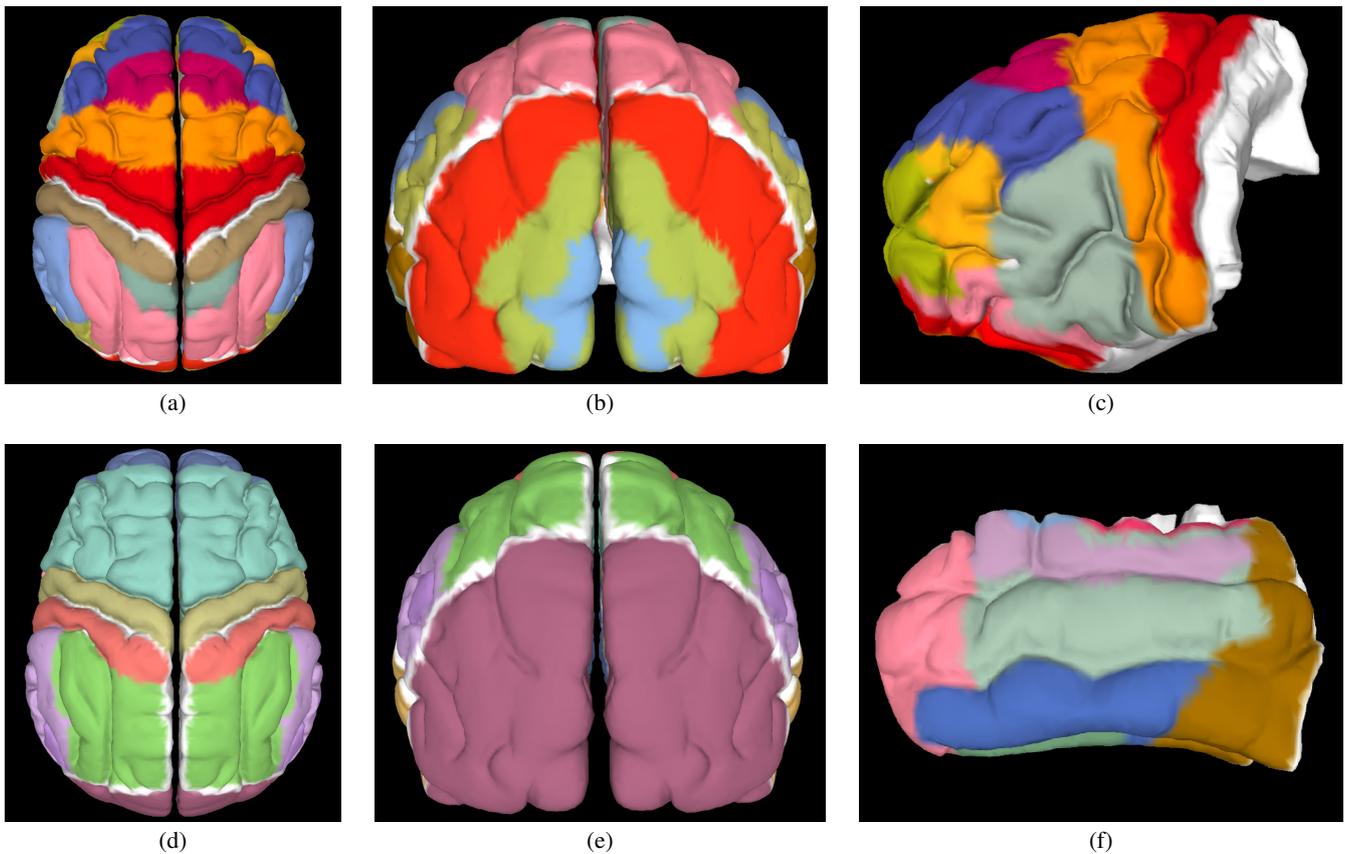
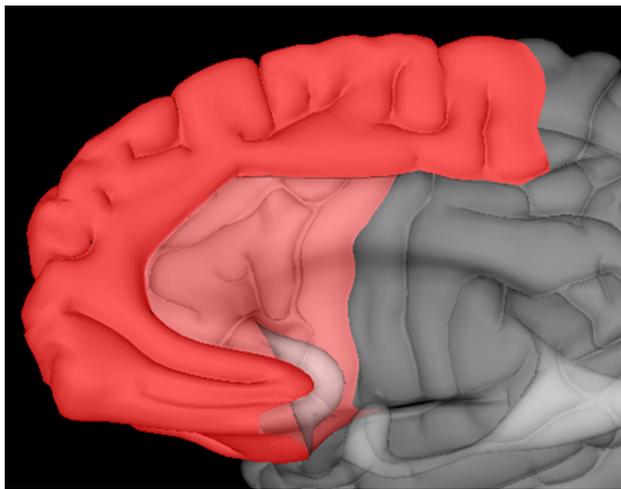
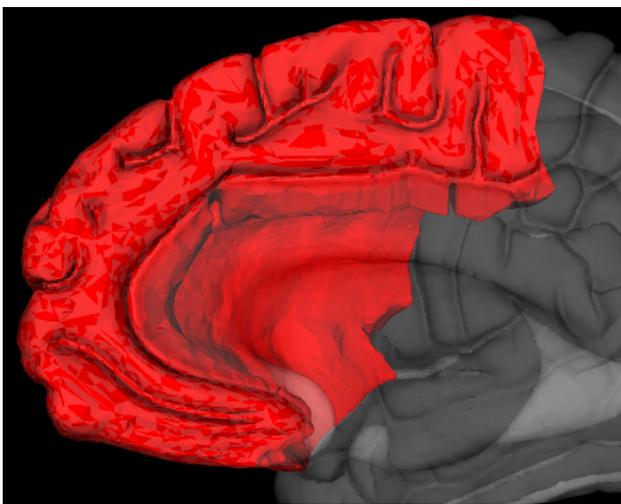


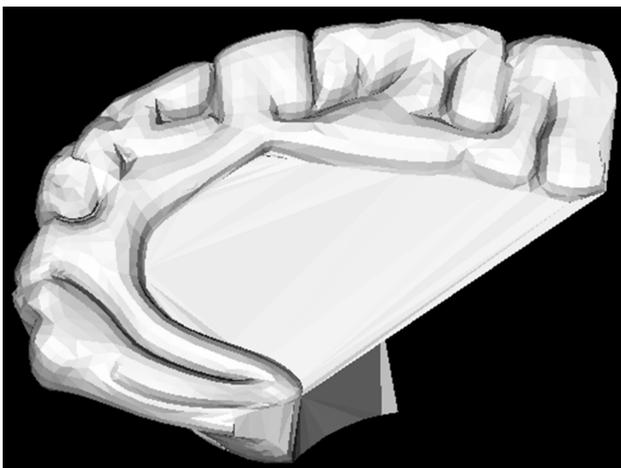
Fig. 9. Axial and frontal view of anatomical classifications *Brodman areas* (a-b) and *neurological centers* (d-e) defined via semi-automated spline definition. The classification is also applicable to sub-parts, like *frontal lobe* (c) or the *temporal lobe* (f) of left hemisphere.



(a)



(b)



(c)

Fig. 8. Based on the marked surface region referring to the frontal lobe of the left hemisphere (a), around 50% of the surface must be complemented to achieve a closed surface mesh representation of the anatomical sub-volume. While the presented randomized dilation based approach leads to good result quality (b), simple border interweaving as provided by like `vtkFillHolesFilter` functionality of VTK is insufficient to approach morphology of the anatomical sub-parts (c).

of various levels of details become feasible. For each sub-mesh, several anatomical classifications can be assigned as marked regions. With the discussed concepts, the classification regions prepared for the surface mesh can be utilized as anatomical classification or as input for sub-mesh generation based on randomized dilation. Thus, topological fragmentation can be achieved at an arbitrary level of granularity.

The discussed spline-based region definition on input mesh representations allows for accurate region specification with minimum user interaction. The 3D spline interaction concept implemented in VTK with support for placing, moving and deleting the spline control points to manipulate the region definitions is applicable to other 3D visualization and interaction frameworks. The triangle selection based on projection borders leads to more predictable results in case of topographical changes, compared to common APSP path completion.

Both, the hierarchical 3D mesh data model as well as the classification functionality is platform independent. The applicability of the 3D environment functionality to mobile devices with altered user interaction is currently evaluated and tested in a subsequent research project.

ACKNOWLEDGMENT

The authors would like to thank the group of software engineering students, developing prototypes in the context of this research work in the course of practical exercises at the University of Applied Sciences Upper Austria, Campus Hagenberg. The project team was composed of *Christoph Absenger, Markus Brindl, Florian Cesal, Josan Pavitter, Reinhard Kaiblinger, Stefan Wurzer* and *Martin Zehetner*.

Special thanks to *Prim.Univ.Prof.Dr. Siegfried G. Priglinger* and *Dipl.-Ing.(FH) Thomas Kaltofen* from the Medical Informatics Group of the RISC research center for support and valuable discussion. Their research project *SEE KID* [13], [12] served as excellent field of application for functional classification of 3D meshes and preparing accurate modeling of human anatomy in the diagnostic domain of strabismus surgery of the eyes.

REFERENCES

- [1] A. Pommert, K.-H. Höhne, B. Pflesser, E. Richter, M. Riemer, T. Schiemann, R. Schubert, U. Schumacher and U. Tiede, *Creating a high-resolution spatial/symbolic model of the inner organs based on the Visible Human*, In: R. Haux and C.A. Kulikowski (eds.): Yearbook of Medical Informatics 2003: Quality of Health Care: The Role of Informatics, Stuttgart, Germany: Schattauer, 2003, 530-537.
- [2] G. Zwettler, W. Backfriedler, R. Swoboda, F. Pfeifer, H. Kratochwill and F. Fellner, *Automatic Liver Classification with Multi-Slice CT Data*, Proc. of the 1st Forschungsforum der österreichischen Fachhochschulen FFH2007, Salzburg, Austria, 2007, 425-426.
- [3] C.-Y. Kao, M. Hofer, G. Sapiro, J. Stern and D. Rottenberg, *A Geometric Method for Automatic Extraction of Sulcal Fundi*, IEEE International Symposium on Biomedical Imaging, 2006, 1168-1171.
- [4] C.B. Barber, D.P. Dobkin and H. Huhdanpaa, *Quickhull Algorithm for Convex Hulls*, ACM Transactions on Mathematical Software (TOMS), 1996.
- [5] W. Schroeder, K. Martin and B. Lorensen, *The Visualization Toolkit An Object-Oriented Approach to 3D Graphics*, 4th ed., Kitware Inc., 2005.
- [6] D. Shreiner, *The Khronos OpenGL ARB Working Group: OpenGL Programming Guide: The Official Guide to Learning OpenGL*, version 4.3, 8th edition, Addison-Wesley, 2013.
- [7] E.W. Dijkstra, *A note on two problems in connexion with graphs*, Numerische Mathematik 1, 1959, 269-271.

- [8] T.M. Liebling, *Voronoi Diagrams and Delaunay Triangulations: Ubiquitous Siamese Twins*, Documenta Mathematica, ISMP, 2012, 419-431.
- [9] G. Zwettler and W. Backfrieder, *Functional Segmentation in 3D Angiography*, Proc. of the 4th Forschungsforum der österreichischen Fachhochschulen FFH2010, Burgenland, Austria, 2010, 79-84.
- [10] S. Hougardy, *The Floyd-Warshall algorithm on graphs with negative cycles*, Information Processing Letters 110, 2010, 279-281.
- [11] K. Brodmann and L.J. Garey, *Brodmann's: Localisation in the Cerebral Cortex*, Springer, 2006.
- [12] T. Kaltofen, G. Santana Sosa and S. Priglinger, *Computer-Based Simulation of the Prism Cover Test with the Biomechanical Eye Model SEE-KID*, Medicine Meets Virtual Reality 21, Manhattan Beach, USA, 2014.
- [13] T. Kaltofen, G. Santana Sosa and S. Priglinger, *SEE-KID - Virtual Ophthalmotrope Versus Biomechanical Model*, 23rd Oculomotor Meeting, Linz, Austria, 2013.
- [14] G. Zwettler, *General Model-Based Segmentation Strategy for Holistic Analysis of Tomographic Medical Image Data in 3D Radiology*, PhD Thesis, University of Vienna, Austria, 2014.
- [15] R.E. Pattis, *EBNF: A Notation to Describe Syntax*, available from <http://www.ics.uci.edu/pattis/ICS-31/lectures/ebnf.pdf> (visited on 07/31/2014), 2013.
- [16] Kitware Inc, *VTK - The Visualization Toolkit*, available from <http://www.vtk.org/> (visited on 10/11/2013), 2013.