Enriching scientific publications with interactive 3D PDF: An integrated toolbox for creating ready-to-publish figures

Axel Newe

Three-dimensional (3D) data of many kinds is produced at an increasing rate throughout all scientific disciplines. The Portable Document Format (PDF) is the de-facto standard for the exchange of electronic documents and allows for embedding three-dimensional models. Therefore, it is a well suited medium for the visualization and the publication of this kind of data. The generation of the appropriate files has been cumbersome so far. This article presents the first release of a software toolbox which integrates the complete workflow for generating 3D model files and ready-to-publish 3D PDF documents for scholarly publications in a consolidated working environment. It can be used out-of-the-box as a simple working tool or as a basis for specifically tailored solutions. A comprehensive documentation, an example project and a project wizard facilitate the customization. It is available royalty-free and for Windows, MacOS and Linux.

1 **Enriching scientific publications with interactive 3D PDF: An**

2 **integrated toolbox for creating ready-to-publish figures**

3 Axel Newe[1,2]

4 [1] Medical Applications Team, Method Park Engineering GmbH, Erlangen, Germany

5 [2] Chair of Medical Informatics, Friedrich-Alexander-University Erlangen-Nuremberg, Erlangen, Germany

7 Corresponding Author:

8 Axel Newe[1,2]

9 Method Park Engineering GmbH, Am Wetterkreuz 19a, 91058 Erlangen, Germany

10 E-mail address: axel.newe@fau.de

Introduction

14 Throughout many scientific disciplines, the availability – and thus the importance – of three-

15 dimensional (3D) data has grown in the recent years. Consequently, this data is often the basis

16 for scientific publications, and in order to avoid a loss of information, the visualization of this

17 data should be 3D whenever possible **(Tory & Möller, 2004)**. In contrary to that, almost all

18 contemporary visualization means (paper printouts, computer screens, etc.) only provide a two-

19 dimensional (2D) interface.

20 The most common workaround for this limitation is to project the 3D data onto the available

21 2D plane **(**Newe, 2015), which results in the so-called "2.5D visualization" **(Tory & Möller,**

22 **2004)**. This projection yields two main problems: limited depth perception and objects that

23 occlude each other. A simple but effective solution of these problems is interaction: by

24 changing the projection angle of a 2.5D visualization (i.e., by changing the point of view), depth

25 perception is improved **(Tory & Möller, 2004)**, and at the same time objects that had previously

26 been occluded (e.g., the backside) can be brought to sight.

27 A means of application of this simple solution has been available for many years: the Portable

28 Document Format (PDF) from Adobe **(Adobe, 2014)**. This file format is the de-facto standard for

29 the exchange of electronic documents and almost every scientific article that is published

30 nowadays is available as PDF – as well as even articles from the middle of the last century

31 **(Hugh-Jones, 1955)**. PDF allows for embedding 3D models and the Adobe Reader

32 (http://get.adobe.com/reader/otherversions/) can be used to display these models

33 interactively.

34 Nevertheless, this technology seems not to have found broad acceptance among the

35 scientific community until now, although journals encourage authors to use this technology

36 **(Maunsell, 2010; Elsevier, 2015)**. One reason might be that the creation of the appropriate

37 model files and of the final PDF documents is still cumbersome. Not everything that is

38 technically possible is accepted by those who are expected to embrace the innovation if the

39 application of this innovation is hampered by inconveniences **(Hurd, 2000)**. Generally suitable

40 protocols and procedures have been proposed by a number of authors before, but they all

41 required of toolchain of at least three **(Kumar et al., 2010**; **Danz & Katsaros, 2011)** or even four

42   **(Phelps, Naeger & Marcovici, 2012; Lautenschlager, 2014)** different software applications and

43   up to 22 single steps until the final PDF was created. Furthermore, some of the proposed

44   workflows were limited to a certain operating system (OS) **(Phelps, Naeger & Marcovici, 2012)**,

45   required programming skills **(Barnes et al., 2013)** or relied on commercial software

46   **(Ruthensteiner & Heß, 2008)**. Especially the latter might be an important limiting factor which

47   hampers the proliferation of the 3D PDF format in scientific publishing **(Lautenschlager, 2014;**

48   **Newe, 2015)**.

49   This article presents a comprehensive and highly integrated software tool for the creation of

50   both the 3D model files (which can be embedded into PDF documents) and the final, ready-to-

51   publish PDF documents with embedded interactive 3D figures. The presented solution is based

52   on MeVisLab, available for all major operating systems (Windows, MacOS and Linux) and

53   requires no commercial license. The source code is available but does not necessarily need to

54   be compiled since binary add-on installers for all platforms are available. A detailed online

55   documentation, an example project and an integrated wizard facilitate re-use and

56   customization.

57   ### *Background and Related Work*

58   ### *The Portable Document Format*

59   The Portable Document Format is a document description standard for the definition of

60   electronic documents independently of the software, the hardware or the operating system

61   that is used for creating or consuming (displaying, printing…) it **(Adobe, 2008a)**. A PDF file can

62   comprise all necessary information and all resources to completely describe the layout and the

63   content of an electronic document, including texts, fonts, images and multimedia elements like

64   audio, movies or 3D models. Therefore, it fulfils all requirements for an interactive publication

65   document as proposed by **(Thoma et al., 2010)**.

66   Although it is an ISO standard (ISO 32000-1:2008 **(ISO, 2008)**), the specification is available to

67   the full extent from the original developer Adobe **(Adobe, 2015)** and can be used royalty-free.

68   ### *Embedding 3D Models into PDF*

69   The fifth edition of the PDF specification (PDF version 1.6 **(Adobe, 2004)**), published in 2004,

70   was the first to support so-called "3D Artwork" as an embedded multimedia feature. In January

71   2005, the Acrobat 7 product family provided the first implementation of tools for creating and

72   displaying these 3D models **(Adobe, 2005)**.

73   The latest version (PDF version 1.7 **(Adobe, 2008a)**) supports three types of geometry (meshes,

74   polylines and point clouds), textures, animations, 15 render modes, 11 lighting schemes and

75   several other features. The only 3D file format that is supported by the ISO standard **(ISO, 2008)**

76   is Universal 3D (U3D, see section below). Support for another 3D format (Product

77   Representation Compact, PRC) has been added by Adobe **(Adobe, 2008b)** and has been

78   proposed to be integrated into the replacement Norm ISO 32000-2 (PDF 2.0). However, this

79   new standard is currently only available as draft version **(ISO, 2014)** and has not yet been

80   adopted.

81   Although the first application in scientific context was proposed in November 2005

82   **(Zlatanova & Verbree, 2005)** and thus quite soon after this new technology was available, it

83   took three more years before the first applications were really demonstrated in scholarly

84   articles **(Ruthensteiner & Heß, 2008**; **Kumar et al., 2008**; **Barnes & Fluke, 2008)**. Since then, the

85   number of publications that apply PDF 3D technology either in theory or in practice has

86   increased almost every year (Table 1). The most sophisticated implementation so far is the

87   reporting of planning results for liver surgery where the PDF roots are hidden behind a user

88   interface which emulates a stand-alone software application **(Newe, Becker & Schenk, 2014)**.

89   **Table 1.** Number of publications related to 3D PDFs in biomedical sciences since 2008 (not

90   comprehensive).

Year	Number of publications with embedded/supplemental 3D PDF	Number of publications dealing with/mentioning 3D PDF
2005	-	1
2008	1	-
2009	5	4
2010	2	7
2011	7	6
2012	6	5
2013	7	2
2014	21	7
2015	31	2

The Universal 3D (U3D) file format

As outlined above, the U3D file format is the only 3D format that is supported by the current ISO specification of PDF. Initially designed as an exchange format for Computer Aided Construction (CAD), it was later standardized by Ecma International (formerly known as European Computer Manufacturers Association, ECMA) as ECMA-363 (Universal 3D File Format). The latest version is the 4th edition from June 2007 **(ECMA, 2007)**.

U3D is a binary file format that comprises all information to describe a 3D scene graph. A U3D scene consists of an arbitrary number of objects that can be sorted in an object tree. The geometry of each object can be defined as a triangular mesh, a set of lines or a set of points. A proprietary bit encoding algorithm allows for a highly compressed storage of the geometry data. A number of additional features and entities (textures, lighting, views, animations) can be defined; details are described in previously published articles **(Newe & Ganslandt, 2013)**.

The scholarly publishing company Elsevier invites authors to supplement their articles with 3D models in U3D format **(Elsevier, 2015)** and many 3D software tools provide the possibility to export in U3D format. However, most of them are commercial software, but open source solutions like MeshLab (http://meshlab.sourceforge.net/) are available as well.

Creating 3D Model Files and PDF Documents

Although many tools and libraries are available that support the creation of 3D model files and of final PDF documents, the whole process is still cumbersome. The problems are manifold: some tools require programming skills; some do not support features those are of interest for scientific 3D data (like polylines **(Newe, 2015)** and point clouds **(Barnes & Fluke, 2008)**). Operating system platform support is another issue, as well as royalty-free use.

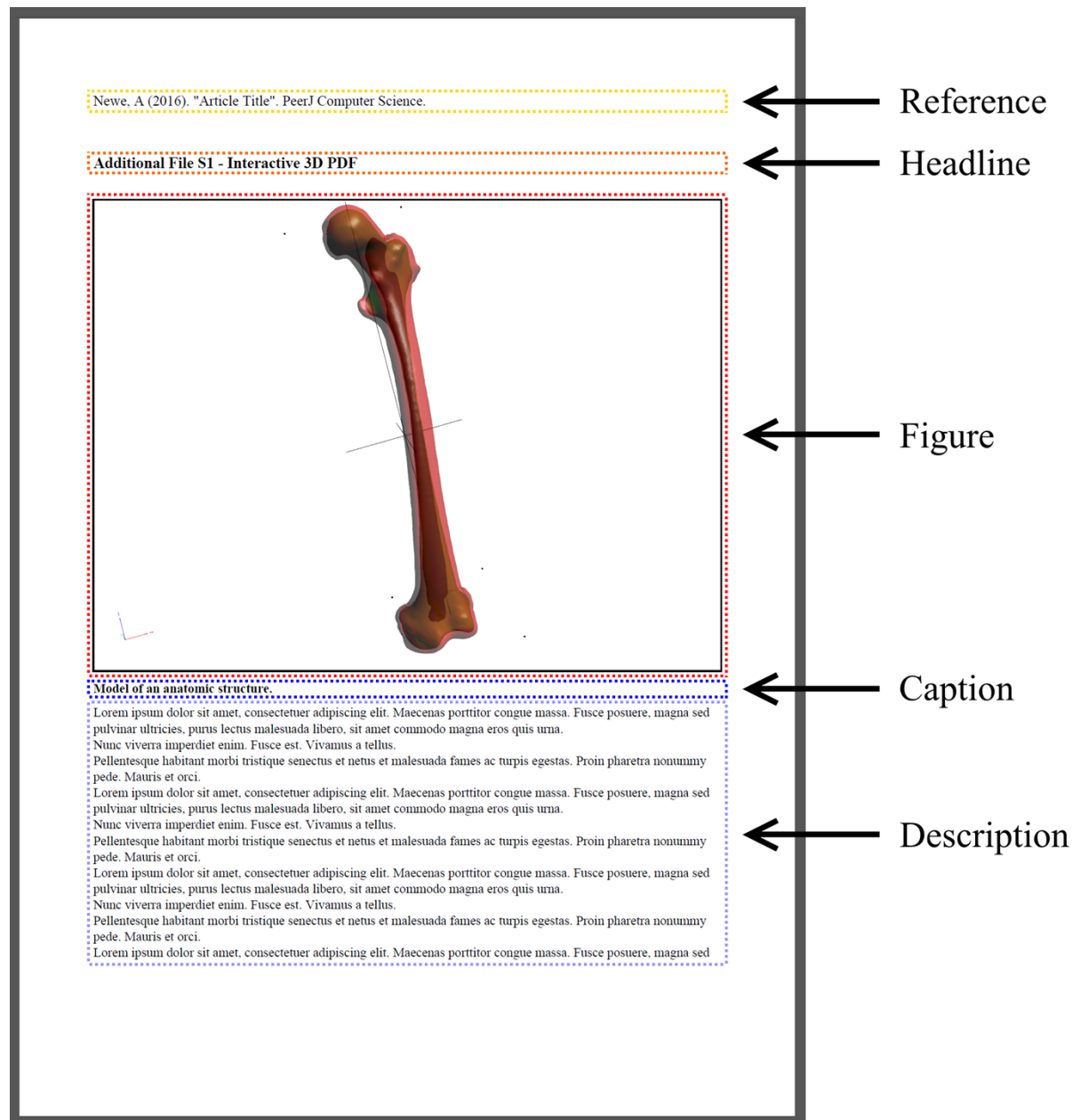
As regards the creation of the 3D model files, most of these problems have been addressed in a previous article **(Newe, 2015)**. The main problem, however, remains the creation of the final PDFs. Specifying the content and (in particular) the layout of a document can be a complex task and is usually the domain of highly specialized word processor software. Figures and supplements for scholarly publications, on the other hand, usually have a specific layout where only the contents of (a limited number of) pre-defined elements vary. There are at least three common elements for a scientific figure: the figure itself, a short caption text and a longer

120   descriptive text. If the figure is intended to be provided as supplemental information file

121   instead of being integrated into the main article text, some additional information is necessary

122   as well: At least a general headline and an optional reference to the main article should be

123   provided. If the document content is modularized to these five key elements (Figure 1), the

124   creation of the PDF itself becomes a rather simple task, because the layout can be pre-defined.

125

126   **Figure 1.** General layout of a scholarly figure if provided as supplemental material.

127   One last difficulty arises from a peculiarity of interactive 3D figures in PDF: the number

128   viewing options (e.g., camera angle, zoom, lighting…) is nearly unlimited. Although such a figure

129   is intended to provide all these options, an author usually wants to define an initial view at the

130   objects, if only to simply ensure that all objects are visible. No freely available tool for PDF

131   creation currently provides a feature to pre-define such a view. The movie15 package for LaTeX

132   **(Grahn, 2005)** provides a mechanism do determine the view parameters, but that requires the

133   generation of intermediate PDFs.

134   Finally it must be mentioned that many previously published 3D models are very large –

135   sometimes up to nearly 100 megabytes **(Krause et al., 2014)**. In most cases, this size could (and

136   should) be reduced significantly, because the density of polygon meshes does usually not need

137   to be very high for illustrative purposes.

138   *MeVisLab*

139   MeVisLab is a framework for image processing and environment for visual development,

140   published by MeVis Medical Solutions AG and Fraunhofer MEVIS in Bremen, Germany. It is

141   available via download (http://www.mevislab.de/download/) for all major platforms (Microsoft

142   Windows, Mac OS and Linux) and has a licensing option which is free for use in non-commercial

143   organizations and research ("MeVisLab SDK Unregistered" license,

144   http://www.mevislab.de/mevislab/versions-and-licensing/). Besides the development features,

145   MeVisLab can be used as a framework for creating sophisticated applications with graphical

146   user interfaces that hide the underlying platform and that can simply be used without any

147   programming knowledge **(Koenig et al., 2006; Heckel, Schwier & Peitgen, 2009; Ritter et al.,**

148   **2011)**. MeVisLab has been evaluated as a very good platform for creating application

149   prototypes **(Bitter et al., 2007)**, is very well documented

150   (http://www.mevislab.de/developer/documentation/) and supported by an active online

151   community (http://www.mevislab.de/developer/community/;

152   https://github.com/MeVisLab/communitymodules/tree/master/Community).

153   All algorithms and functions included into MeVisLab are represented and accessed by

154   "modules", which can be arranged and connected to image processing networks or data

155   processing networks on a graphical user interface (GUI) following the visual data-flow

156   development paradigm. By means of so-called "macro modules", these networks can then be

157   converted with little effort into complete applications with an own GUI.

Methods

Elicitation of Requirements

160   As described above, the generation of the necessary 3D model data and particularly of the

161   final PDF is still subject to a number of difficulties. Therefore, the first step was the creation of a

162   list of requirements specifications with the aim to create a tool that overcomes these known

163   drawbacks.

164   **Table 2.** Requirements for the development of the software tool. The two main requirements are

165   highlighted in bold font.

ID	Requirement Specification
R1	**The software *shall* create ready-to-publish PDF documents with embedded 3D models.**
R1.1	The software *shall* offer an option to specify the activation mode and the deactivation mode for the 3D models.
R2	**The software *shall* provide an integrated, single-window user interface that comprises all necessary steps.**
R3	The software *shall* be executable under Windows, MacOS and at least one Linux distribution.
R4	The software *shall* be executable without the need to purchase a commerical license.
R5	The software *shall* create 3D model files in U3D format.
R5.1	The software *shall* create view definitions for the 3D model.
R5.2	The software *shall* create poster images for the PDF document.
R6	The software *shall* import mesh geometry from files in OBJ, STL and PLY format.
R6.1	The software *should* import mesh geometry from other file formats as well.
R6.2	The software *shall* offer an option to reduce the number of triangles of imported meshes.
R6.3	The software *shall* offer an option to specify the U3D object name and the color of imported meshes.
R7	The software *shall* import line set geometry from files in text format.
R7.1	The software *shall* offer an option to specify the U3D object name and the color of imported line sets.
R8	The software *shall* import point set geometry from files in text format.
R8.1	The software *shall* offer an option to specify the U3D object name of imported point sets.

166   Two requirements have been identified to be the most important ones: 1) the demand for a

167   tool that creates "ready-to-publish" PDF documents without the need for commercial software

168   and 2) the integration of all necessary steps into a single and easy-to-use interface. Besides

169   these two main requirements, a number of additional requirements have then been identified

170   as well. See Table 2 for a full list of all requirements that were the basis for the following

171   development.

Creation of an "App" for MeVisLab

172
173 MeVisLab-based solutions presented in previous work **(Newe & Ganslandt, 2013; Newe,**

174 **2015)** already provide the possibility to create U3D files without requiring programming skills

175 and without the need for an intensive training. However, they still needed some basic training

176 as regards assembling the necessary processing chains in MeVisLab. Furthermore, the creation

177 of the final PDF was not possible so far.

178 Therefore, a new macro module was created for MeVisLab. A macro module encapsulates

179 complex processing networks and can provide an integrated user interface. In this way, the

180 internal processes can be hidden away from the user, who can focus on a streamlined workflow

181 instead. Designed in an appropriate way, a macro module can also be considered as an "app"

182 inside of MeVisLab.

183 In order to provide the necessary functionality, some auxiliary tool modules (e.g., for the

184 creation of the actual PDF file) needed to be developed as well. Along with the modules for U3D

185 export mentioned above, these auxiliary tool modules were integrated into the internal

186 processing network of the main "app" macro. The technical details of these internal modules

187 are not within the scope of this article. However, the source code is available and interested

188 readers are free to explore the code and to use it for own projects.

189 The user interface of the app was designed in a way that it guides novice users step-by-step

190 without treating experienced users too condescendingly, though. Finally, a comprehensive

191 documentation including an example project, a wizard for creating tailored PDF modules and a

192 verbose help text was set up.

Deployment of Core Functionality

193
194 For the creation of the actual PDF files, version 2.2.0 of the cross-platform, open source

195 library libHaru (http://libharu.org/) was selected, slightly modified and integrated as third-party

196 contribution into MeVisLab.

197 Next, the application programming interface (API) of libHaru was wrapped into an abstract

198 base module for MeVisLab in order to provide an easy access to all functions of the library and

199 in order to hide away standard tasks like creating a document or releasing memory. A large

200 number of convenience functions were added to this base module and an exemplary MeVisLab

201   project was set up in order to demonstrate how to use the base module for tailored

202   applications. This base module also served as basis for the PDF creation of the app macro

203   described above. Finally, a project wizard was integrated into the MeVisLab GUI.

Results

The "Scientific3DFigurePDFApp" module

206   The new macro module "Scientific3DFigurePDFApp" for MeVisLab provides an integrated

207   user interface for all steps that are necessary for the creation of U3D models files and for the

208   creation of the final PDF documents with embedded 3D models. The model editor part

209   produces U3D model files of geometry data that are compatible with version 4 of the ECMA-

210   363 standard and poster images in Portable Network Graphics (PNG) format. The PDF editor

211   part produces PDF documents that are compliant with PDF version 1.7 (ISO 32000-1:2008). An

212   example PDF is available as Supplemental File S1.

213   The user interface is arranged in tabs, whereas each tab comprises all functions for one step

214   of the workflow. By processing the tabs consecutively, the user can assemble and modify 3D

215   models, save them in U3D format, create views and poster images for the PDF document, and

216   finally create the PDF itself step by step (Figure 2).

217   The raw model data can be collected in two ways: either by feeding it to the input connectors

218   or by assembling it by means of the built-in assistant. The former option is intended for

219   experienced MeVisLab users that want to attach the module at the end of a processing chain.

220   The latter option addresses users that simply want to apply the app for converting existing 3D

221   models and for creating an interactive figure for scholar publishing.

222

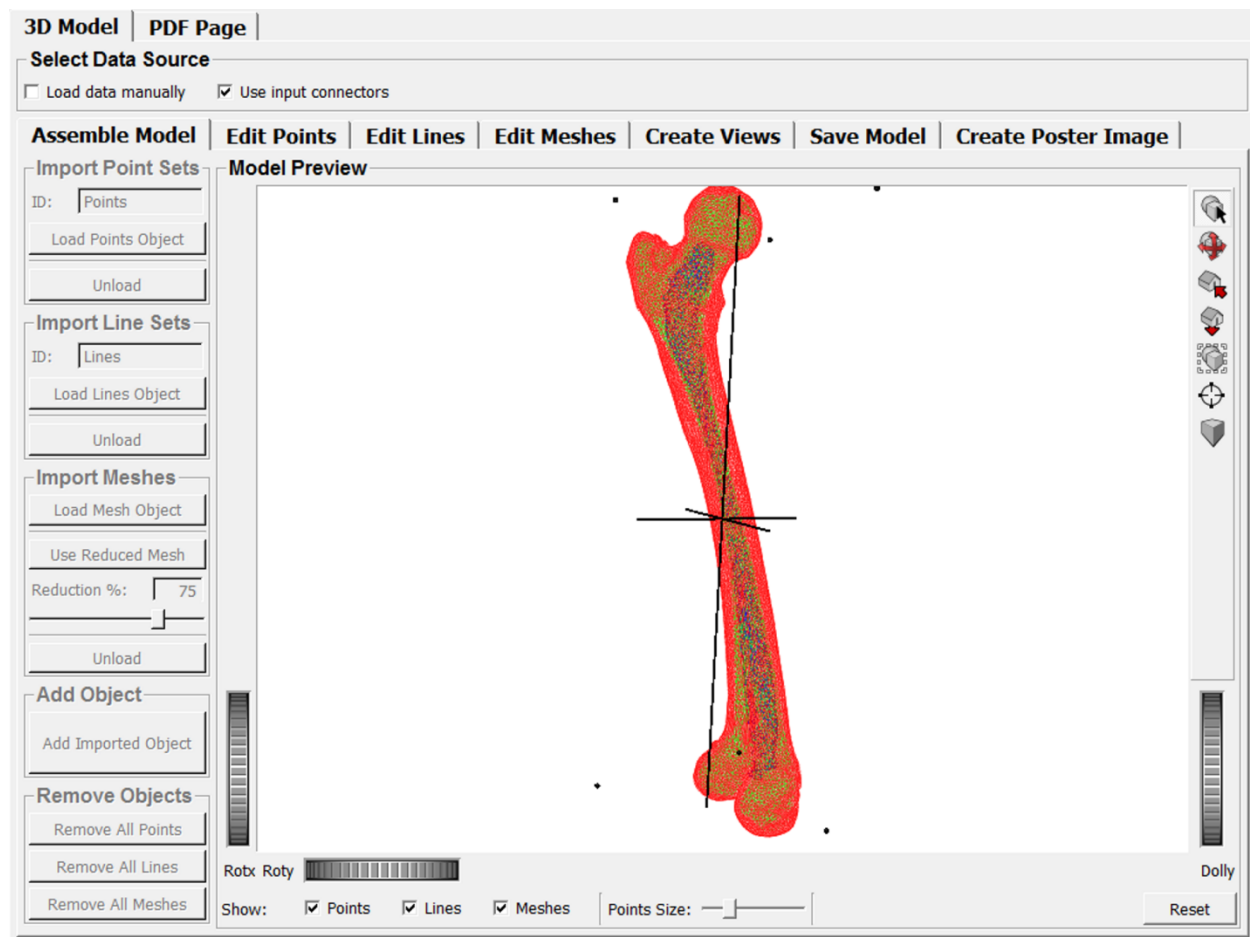



Figure 2. User interface of the app. The user interface comprises all necessary steps for the creation of 3D model files and PDF files. It is arranged in tabs for each step.

The software allows for importing the geometry data of 39 different 3D formats, including point clouds and line sets from files in character-separated value (CSV) format (see Table 3 for a full list). The import of textures and animations is not supported.

Objects from different sources can be combined and their U3D properties (colour, name, position in the object tree) can be specified. The density of imported meshes can be adjusted interactively and multiple views (i.e., the specification of camera, lighting and render mode) can be pre-defined interactively as well. Finally, it is also possible to create a poster image which can replace an inactive 3D model in the PDF document if the model itself is disabled or if it cannot be displayed for some reason (e.g., because the reading software does not provide the necessary features).

235   **Table 3.** List of supported 3D formats for importing geometry data (textures, animations and other
236   features are not supported).

File format	Typical File Extension(s)
Stereolithography	*.stl
Stanford Polygon Library	*.ply
Wavefront Object	*.obj
Object File Format	*.off
Blender	*.blend
Raw Triangles	*.raw
Raw Point Clouds	*.csv; *.txt
Raw Line Sets	*.csv; *.txt
3D GameStudio	*.mdl; *.hmp
3D Studio Max	*.3ds; *.ase
AC3D	*.ac
AutoCAD/Autodesk	*.dxf
Biovision BVH	*.bvh
CharacterStudio Motion	*.csm
Collada	*.dae; *.xml
DirectX X	*.x
Doom 3	*.md5mesh; *.md5anim; *.md5camera
Irrlicht	*.irrmesh; *.irr;*.xml
LightWave	*.lwo; *.lws
Milkshape 3D	*.ms3d
Modo Model	*.lxo
Neutral File Format	*.nff
Ogre	*.mesh.xml, *.skeleton.xml, *.material
Quake I, Quake II, Quake III	*.mdl; *.md2; *.md3; *.pk3
Quick3D	*.q3o; *q3s
RtCW	*.mdc
Sense8 WorldToolkit	*.nff
Terragen Terrain	*.ter
TrueSpace	*.cob, *.scn
Valve Model	*.smd, *.vta
XGL	*.xgl, *.zgl

237   All functions are explained in detail in a comprehensive documentation which can be

238   accessed directly inside MeVisLab. A stand-alone copy of the documentation is available as

239   Supplementary File S2. In order to use the app, it simply needs to be instantiated (via the

240   MeVisLab menu: Modules → PDF → Apps → Scientific3DFigurePDFApp). A full feature list is

241   available in Table 4.

242   **Table 4.** List of features.

Category	Features
Data Import	Import external data, import MeVisLab data, import point clouds, import line sets, import meshes from 37 file formats, adjust mesh density, preview import
Point Cloud Editing	Specify point cloud name, specify position in model tree, preview settings
Line Set Editing	Specify line set name, specify position in model tree, specify colour, preview settings
Mesh Editing	Specify mesh name, specify position in model tree, specify colour, specify opacity, preview settings
View Specification	Specify view name, specify background colour, specify lighting scheme, specify render mode, preview settings, specify multiple views
U3D Creation	Store model in U3D format, preview scene
Poster Image Creation	Store poster in PNG format, preview scene, specify superimposed text
PDF Creation	Store document in PDF (v1.7) format, specify header citation text, specify header headline text, specify U3D file, specify poster file, specify model activation mode, specify model deactivation mode, specify toolbar enabling, specify navigation bar enabling, specify animation start mode, specify caption, specify description text

243   ***Additional Features for Tailored PDF Creation***

244   The abstract module which wraps the API of the PDF library libHaru into a MeVisLab module

245   was made public ("PDFGenerator" module) and can be used for the development of tailored

246   MeVisLab modules. In order to facilitate the re-use of this abstract base module, an exemplary

247   project was set up (/Projects/PDFExamples/SavePDFTemplate). This project demonstrates how

248   to derive a customized module from the PDFGenerator base module and how to specify the

249   content of the PDF file that will be created by means of the new module. The template code is

250   verbosely annotated and includes examples for setting PDF properties (e.g., meta data, page

251   size, encryption) as well as the document content (including text, images, graphics and 3D

252   models). The output of the SavePDFTemplate module is illustrated in Figure 3.

253

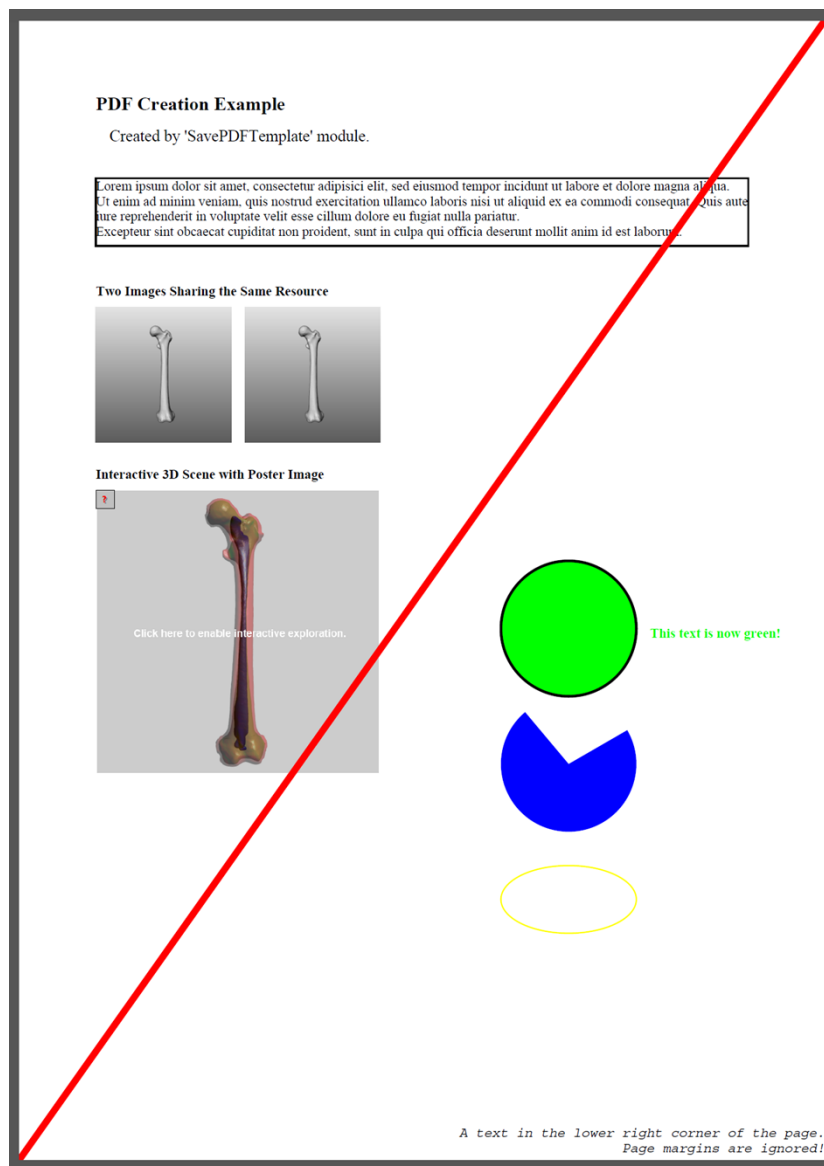



Figure 3. Output of the SavePDFTemplate module.

255   Finally, a project wizard was integrated into the MeVisLab GUI. It can be accessed via the

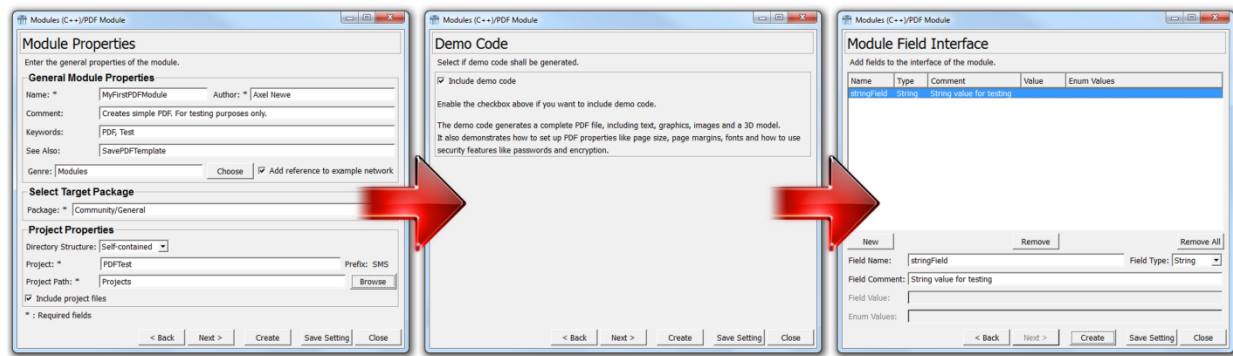
256   MeVisLab menu: File → Run Project Wizard… → PDF Module. The wizard consists of three steps

257   (Figure 4) whereof the second step offers the possibility to include demo code which produces

258   the same PDF file as the SavePDFTemplate module described above. The general usage of

259   project wizards in MeVisLab is explained in chapter 23 of the MeVisLab manual (menu: Help →

260   Browse Help Pages → Using MeVisLab → MeVisLab Reference Manual).

261

262   **Figure 4.** Project wizard for creating customized PDF modules.

263   ***Availability***

264   The whole PDF project for MeVisLab (which includes the Scientific3DFigurePDFApp, the

265   PDFGenerator base module, the SavePDFTemplate project, the project wizard and all source

266   code files) is available for Microsoft Windows, MacOS and Linux (tested with Ubuntu 14.04.2). It

267   requires MeVisLab 2.8 or a later version (http://www.mevislab.de/download/). The Windows

268   version of MeVisLab is available for four compiler versions. This is relevant only if the source

269   code is intended to be compiled manually (see below).

270   There are two approaches to add the app and the other elements to an existing MeVisLab

271   installation: add-on installers and the online repository of the MeVisLab community sources.

272   Installers are self-contained, executable archives that automatically add all necessary files to

273   an existing MeVisLab installation. The target groups for these installers are MeVisLab

274   newcomers and pure users that want to use the Scientific3DFigurePDFApp out-of-the-box. The

275   current version of the installers for all operating systems and all 64-Bit compiler versions can be

276   downloaded   from   the   research   data   repository   Zenodo

277   (http://dx.doi.org/10.5281/zenodo.48758). Installers for the previous version of MeVisLab

278   (2.7.1) are available as well (http://dx.doi.org/10.5281/zenodo.47491), but this version will not

279   be supported in the future. Updates will be made available via Zenodo as well. A dedicated

280   Zenodo Community Collection named "Three-dimensional Portable Document Format (3D PDF)

281   in Science" has been set up for this purpose (https://zenodo.org/collection/user-3d-pdf).

282   All those who are interested in being able to always use the latest version should connect

283   their MeVisLab installation with the community sources which are hosted at GitHub

284   (https://github.com/MeVisLab/communitymodules/tree/master/Community).   This   approach,

285   however, requires compiling the source code and is intended only for experienced users or for

286   users that are willing to become acquainted with MeVisLab. Note, that there are multiple

287   versions available for Windows, depending on the compiler that is intended to be used.

288   *Discussion*

289   *A Toolbox for the Creation of 3D PDFs*

290   The utilization of 3D PDF technology for scholarly publishing has been revealed and proven

291   both useful and necessary by several authors in the past years. The mainstream application of

292   3D PDF in science, however, is yet to come.

293   One reason might be the difficult process that has so far been necessary to create

294   appropriate data and relevant electronic documents. This article presents an all-in-one solution

295   for the creation of such files which requires no extraordinary skills. It can be used by low-end

296   users as an out-of-the-box tool as well as a basis for sophisticated tailored solutions for high-

297   end users.

298   Many typical problems as regards the creation of 3D model files have been addressed and

299   solved. All steps of the workflow are integrated seamlessly. The software is available for all OS

300   platforms and can import and process objects from many popular 3D formats, including

301   polylines and point clouds (Table 3). The density of imported meshes can be adjusted

302   interactively which enables the user to find the best balance between the desired level of detail

303   and the file size.

304   The main contribution, however, is the possibility to create ready-to-publish PDF documents

305   with a minimum of steps. This approach was proposed to be the ideal solution by **(Kumar et al.,**

306   **2010)**. To best knowledge, this is the first time that such an integrated and comprehensive

307   solution is made available for the scientific community.

308   *Applications*

309   The areas of application (see an example in Supplemental File S1) are manifold and not

310   limited to a specific scientific discipline. On the contrary: every field of research that produces

311   three-dimensional data can and should harness this technology in order to get the best out of

312 that data.

313 One (arbitrary) example for the possible use of mesh models from the recent literature is 3D

314 ultrasound. Dahdouh et.al. recently published about the results of segmentation of obstetric 3D

315 ultrasound images **(Dahdouh et al., 2015)**. That article contains several figures that project

316 three-dimensional models on the available two-dimensional surface. A presentation in native

317 3D would have enabled the reader to interactively explore the whole models instead of just one

318 pre-defined snapshot. Another example is the visualization of molecular structures as

319 demonstrated by **(Kumar et al., 2008)**.

320 Polylines can be used to illustrate nervous fibre tracking. **(Mitter et al., 2015)** used 2D

321 projections of association fibres in the foetal brain to visualize their results. A real 3D

322 visualization would have been very helpful in this case as well: While some basic knowledge

323 about a depicted object helps to understand 2D projections of 3D structures, the possibility to

324 preserve at least a little depth perception decreases with an increasing level of abstraction

325 (mesh objects vs. polylines).

326 This particularly applies to point clouds which can be observed, for example, in an article by

327 **(Qin et al., 2015)**: Although these authors added three-dimensional axes to their figure (no. 6) it

328 is still hard to get an impression of depth and therefore of the real position of the points in 3D

329 space.

330 ### *Limitations*

331 Although the presented software pulls down the major thresholds that impede the creation

332 of interactive figures for scholarly publishing, some limitations still need to be considered.

333 A general concern is the suitability of PDF as a means to visualize and to exchange 3D models.

334 PDF and U3D (or PRC) do not support all features that other modern 3D formats provide and

335 that would be of interest for the scientific community (e.g., volumetric models). On the other

336 hand, PDF is commonly accepted and de-facto the only file format that is used for the

337 electronic exchange of scholarly articles. Therefore, PDF may not be the perfect solution, but it

338 is the best solution that is currently available.

339 The presented software requires MeVisLab as background framework and the installation of

340 MeVisLab requires a medium-sized download of about 1 GB (depending on the operating

341   system), which could be considered rather large for a PDF creator. On the other hand,

342   MeVisLab integrates a large library for the processing and the visualization of (biomedical)

343   image data. Furthermore, other frameworks (like MeshLab) do not provide all necessary

344   features (e.g., polylines or point clouds) and therefore were not considered to meet basic

345   requirements for the development of the software tool.

346   The import of 3D models is based on the Open Asset Import Library

347   (*http://www.assimp.org/*) which does not support all features of all 3D formats. For example,

348   textures and animations cannot be imported and should thus not be embedded into a model

349   file that is intended to be imported. However – although the model-editor part of the

350   presented software does not support textures (or animations), the PDF-creator part can still be

351   used to produce scientific PDFs with textured or animated models, if the necessary U3D files

352   have been created with external and more specialized software. In this use case, the

353   Scientific3DFigurePDFApp does not integrate all necessary steps, but it still remains a "few-

354   clicks" alternative for the creation of interactive PDF supplements for scientific publications and

355   it still obviates the need for a commercial solution.

356   Finally, very large model files should be avoided. If a large model fails to import, it should be

357   separated into several sub-models. A mesh reduction can be applied after the import, but a

358   previously reduced mesh speeds up the import process.

359   ### *Suitable Reading Software*

360   The Adobe Reader (http://get.adobe.com/reader/otherversions/) is available free of charge

361   for all major operating systems (MS Windows, Mac OS, Linux). It is currently the only software

362   that can be used to display embedded 3D models and to let the user interact with them

363   (zooming, panning, rotating, selection of components). However, even the Adobe Reader does

364   not support all U3D features **(Adobe, 2007)**, e.g., Glyphs and View Nodes. Furthermore, a

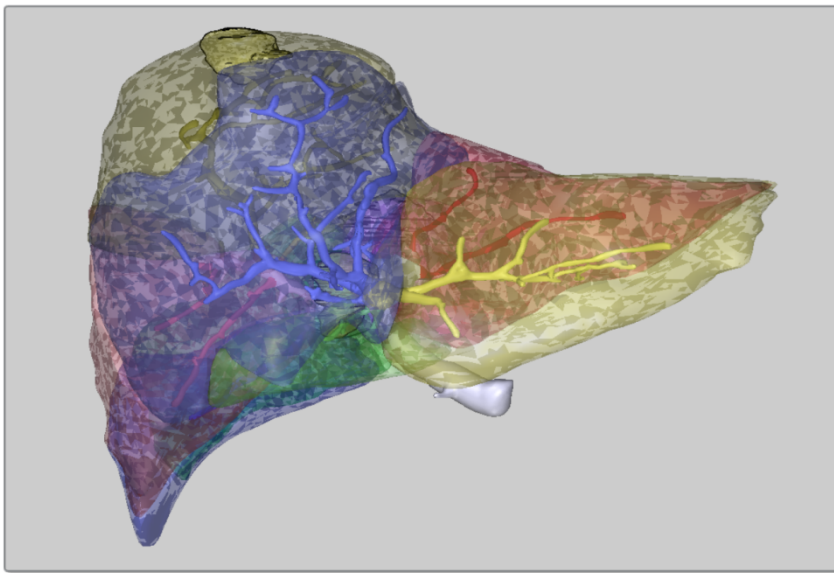
365   rendering flaw has been observed on low-end graphic boards in MacOS hardware (Figure 5).

366   Adobe Reader for MacOS does not render transparent surfaces superimposed upon each other

367   correctly: instead, a strong tessellation effect is visible. This may also occur on other platforms

368   but has not been reported yet. Since this is an issue with the rendering engine of Adobe Reader,

369   there is currently no other solution than using a different render mode (e.g., one of the

370 wireframe modes) or different hardware.

371

372 **Figure 5. Rendering artifacts.** These tessellation artifacts have been observed on MacOS systems with
373 low-end graphic hardware.

374   Experience shows that many users do not expect a PDF document to be interactive.

375 Therefore, possible consumers should be notified that it is possible to interact with the

376 document and they should also be notified that the original Adobe Reader is required for this.

377 Although poster images are a workaround to avoid free areas in PDF readers that are not

378 capable of rendering 3D scenes, missing 3D features of a certain reader could be confusing for a

379 user.

380 ### *A Basis for Own Modules*

381   As pointed out in previous work **(Newe, 2015)**, the authoring of a PDF document is usually a

382 complex task and thus in most cases it cannot be avoided to separate the generation of 3D

383 model data from the actual PDF authoring. Although the software tool presented in this article

384 mitigates this general problem by integrating model generation and PDF creation, it is still

385 limited to a certain use case and a pre-defined PDF layout.

386   However, the API of the core PDF functionality is public and designed in a way that facilitates

387 the creation of own PDF export modules. The large number of convenience functions for the

388 abstract base module (PDFGenerator) facilitates the creation of derived modules. These

389 functions massively lighten the programmer's workload by providing a simple access to routine

390 tasks like writing text at a defined position or like embedding a 3D model which would normally

391 require a whole series of API calls. Finally, the built-in wizard generates all necessary project

392 files and source code files to create a fully functional module barebone which only needs to be

393 outfitted with the desired functionality.

394 ***Outlook***

395 Although this article represents an important milestone, the development of the PDF project

396 for MeVisLab is ongoing. Future goals are the integration of virtual volume rendering **(Barnes et**

397 **al., 2013)**, animations **(van de Kamp et al., 2014)** and the parsing of U3D files that have been

398 created with external software. The progress can be tracked via GitHub

399 (https://github.com/MeVisLab/communitymodules/tree/master/Community) and updates to

400 the binary files will be published regularly.

401 ***Conclusion***

402 Three-dimensional data is produced at an increasing rate throughout all scientific disciplines.

403 The Portable Document Format is a well suited medium for the visualization and the publication

404 of this kind of data. With the software presented in this article, the complete workflow for

405 generating 3D model files and 3D PDF documents for scholarly publications can be processed in

406 a consolidated working environment, free of license costs and with all major operating systems.

407 The software addresses novices as well as experienced users: On the one hand, it provides an

408 out-of-the-box solution that can be used like a stand-alone application, and on the other and all

409 sources and APIs are freely available for specifically tailored extensions.

410

411   *List of abbreviations*

412   2D – Two-dimensional

413   3D – Three-dimensional

414   PDF – Portable Document Format

415   ISO – International Organization for Standardization

416   U3D – Universal 3D

417   PRC – Product Representation Compact

418   CAD – Computer Aided Construction

419   ECMA – European Computer Manufacturers Association

420   GUI – Graphical User Interface

421   API – Application Programming Interface

422   PNG – Portable Network Graphics

423   CSV – Character-separated Value, Comma-separated Value

424

425   *Acknowledgements*

426   Thanks to Dr. Hans Meine, Fraunhofer MEVIS, Bremen, Germany, for compiling the code for

427   MacOS and Linux and for building installers.

428   Thanks to Dr. Thomas van de Kamp, Karlsruhe Institute of Technology, Karlsruhe, Germany, for

429   providing test data.

430

431 ***References***

432 **Adobe Systems Incorporated (Adobe).** 2004. PDF Reference, fifth edition, Adobe Portable Document

433 Format, Version 1.6. *Available at*

434 *http://wwwimages.adobe.com/www.adobe.com/content/dam/Adobe/en/devnet/pdf/pdfs/pdf_referenc*

435 *e_archives/PDFReference16.pdf* (accessed: 2016-05-01). Archived at:

436 http://www.webcitation.org/6gUnINDkD.

437 **Adobe Systems Incorporated (Adobe).** 2005. Adobe Announces Acrobat 7.0 Software Availability.

438 *Available at*

439 *https://www.adobe.com/aboutadobe/pressroom/pressreleases/200501/010505Acrobat.html* (accessed:

440 2016-05-01). Archived at: http://www.webcitation.org/6gUnQPxOe.

441 **Adobe Systems Incorporated (Adobe).** 2007. U3D Supported Elements. *Available at*

442 *http://www.adobe.com/content/dam/Adobe/en/devnet/acrobat/pdfs/U3DElements.pdf* (accessed:

443 2016-05-01). Archived at: http://www.webcitation.org/6eU07DNbj.

444 **Adobe Systems Incorporated (Adobe).** 2008. Document management — Portable document format —

445 Part 1: PDF 1.7. Available at

446 http://wwwimages.adobe.com/content/dam/Adobe/en/devnet/pdf/pdfs/PDF32000_2008.pdf

447 (accessed: 2016-05-01). Archived at: http://www.webcitation.org/6S8xkrvPw.

448 **Adobe Systems Incorporated (Adobe).** 2008. Adobe® Supplement to the ISO 32000, BaseVersion: 1.7,

449 ExtensionLevel: 3. *Available at*

450 *http://wwwimages.adobe.com/content/dam/Adobe/en/devnet/pdf/pdfs/adobe_supplement_iso32000.*

451 *pdf* (accessed: 2016-05-01). Archived at: http://www.webcitation.org/6gUnW4iyd.

452 **Adobe Systems Incorporated (Adobe).** 2014. Adobe - PDF: The Global Standard For Trusted, Electronic

453 Documents and Forms. *Available at http://www.adobe.com/au/pdf/about/history/* (accessed: 2016-05-

454 01). Archived at: http://www.webcitation.org/6eTx1aRad.

455 **Adobe Systems Incorporated (Adobe).** 2015. PDF Reference and Adobe Extensions to the PDF

456 Specification. *Available at http://www.adobe.com/devnet/pdf/pdf_reference.edu.html* (accessed: 2016-

457 05-01). Archived at: http://www.webcitation.org/6S8y0ShjG.

458 **Barnes DG, Fluke CJ.** 2008. Incorporating interactive 3-dimensional graphics in astronomy research

459 papers. *New Astronomy*, 13(8):599-605. DOI: 10.1016/j.newast.2008.03.008.

460 **Barnes DG, Vidiassov M, Ruthensteiner B, Fluke CJ, Quayle MR, McHenry CR.** 2013. Embedding and

461 Publishing Interactive, 3-Dimensional, Scientific Figures in Portable Document Format (PDF) Files. *PLoS*

462 *ONE*, 8(9):e69446. DOI: 10.1371/journal.pone.0069446. PMID: 24086243.

463 **Bitter I, Van Uitert R, Wolf I, Ibanez L, Kuhnigk JM.** 2007. Comparison of four freely available

464 frameworks for image processing and visualization that use ITK. *IEEE Transactions on Visualization and*

465 *Computer Graphics*, 13(3):483-493. DOI: 10.1109/TVCG.2007.1001. PMID: 17356215.

466   **Dahdouh S, Angelini ED, Grangé G, Bloch I.** 2015. Segmentation of embryonic and fetal 3D ultrasound
467   images based on pixel intensity distributions and shape priors. *Medical Image Analysis*, 24(1):255-268.
468   DOI: 10.1016/j.media.2014.12.005. PMID: 25655408.

469   **Danz JC, Katsaros C.** 2011. Three-dimensional portable document format: a simple way to present 3-
470   dimensional data in an electronic publication. *American Journal of Orthodontics and Dentofacial*
471   *Orthopedics*, 140(2):274-276. DOI: 10.1016/j.ajodo.2011.04.010. PMID:21803267.

472   **ECMA International (ECMA).** 2007. Standard ECMA-363, Universal 3D File Format, 4th edition (June
473   2007). *Available at http://www.ecma-international.org/cgi-bin/counters/unicounter.pl?name=ECMA-*
474   *363_4thedition&deliver=http://www.ecma-international.org/publications/files/ECMA-ST/ECMA-*
475   *363%204th%20edition.pdf* (accessed: 2016-05-01). Archived at:
476   http://www.webcitation.org/6QkI44nqn.

477   **Elsevier.** 2015. Interactive U3D models. *Available at http://www.elsevier.com/about/content-*
478   *innovation/u3d-models* (accessed: 2016-05-01). Archived at: http://www.webcitation.org/6eTxA8hgV.

479   **Grahn A.** 2005. The movie15 Package. *Available at*
480   *http://solar.physics.montana.edu/kankel/ph567/resources/LaTeX/graphics-animations/movie15.pdf*
481   (accessed: 2016-05-01). Archived at: http://www.webcitation.org/6eTzfqj6v.

482   **Heckel F, Schwier M, Peitgen HO.** 2009. Object oriented application development with MeVisLab and
483   Python. In *Lecture Notes in Informatics (Informatik 2009: Im Focus das Leben), Beiträge der 39.*
484   *Jahrestagung der Gesellschaft für Informatik e.V. (GI), 28.9. - 2.10.2009 in Lübeck*. 1338-1351. ISBN: 978-
485   3-88579-248-2.

486   **Hugh-Jones P.** 1955. Diabetes in Jamaica. *The Lancet*, 266(6896):891-897. DOI: 10.1016/S0140-
487   6736(55)92530-7. PMID: 13264638.

488   **Hurd JM.** 2000. The Transformation of Scientific Communication: A Model for 2020. *Journal of the*
489   *American Society for Information Science and Technology*, 51(14):1279-1283. DOI: 10.1002/1097-
490   4571(2000)9999:99993.0.CO;2-1.

491   **International Organization for Standardization (ISO).** 2008. ISO 32000-1:2008 Document management -
492   - Portable document format -- Part 1: PDF 1.7. *Available at*
493   *http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=51502* (accessed:
494   2016-05-01). Archived at: http://www.webcitation.org/6QkFZgVk9.

495   **International Organization for Standardization (ISO).** 2014. ISO/DIS 32000-2 Review Guide. *Available at*
496   *http://www.pdfa.org/wp-content/uploads/2014/09/ISO-32000-2_DIS-1_REVIEW_ONLY.pdf* (accessed:
497   2016-05-01). Archived at: http://www.webcitation.org/6byQJVKAs.

498   **Koenig M, Spindler W, Rexilius J, Jomier J, Link F, Peitgen HO.** 2006. Embedding VTK and ITK into a
499   visual programming and rapid prototyping platform. In *Proceedings of SPIE, Medical Imaging 2006:*
500   *Visualization, Image-Guided Procedures, and Display; San Diego*. 61412O. DOI: 10.1117/12.652102.

501 **Krause DW, Wible JR, Hoffmann S, Groenke JR, O'Connor PM, Holloway WL, Rossie JB.** 2014.
502 Craniofacial Morphology of Vintana Sertichi (Mammalia, Gondwanatheria) from the Late Cretaceous of
503 Madagascar. *Journal of Vertebrate Paleontology*, 34(Supplement 1):14-109. DOI:
504 10.1080/02724634.2014.976129.

505 **Kumar P, Ziegler A, Grahn A, Hee CS, Ziegler A.** 2010. Leaving the structural ivory tower, assisted by
506 interactive 3D PDF. *Trends in Biochemical Sciences*, 35(8):419-422. DOI: 10.1016/j.tibs.2010.03.008.
507 PMID: 20541422.

508 **Kumar P, Ziegler A, Ziegler J, Uchanska-Ziegler B, Ziegler A.** 2008. Grasping molecular structures
509 through publication-integrated 3D models. *Trends in Biochemical Sciences*, 33(9):408-412. DOI:
510 10.1016/j.tibs.2008.06.004.

511 **Lautenschlager S.** 2014. Palaeontology in the third dimension: a comprehensive guide for the
512 integration of three-dimensional content in publications. *Paläontologische Zeitschrift*, 88(1).
513 DOI:10.1007/s12542-013-0184-2.

514 **Maunsell J.** 2010. Announcement Regarding Supplemental Material. *Journal of Neuroscience*,
515 30(32):10599-10600.

516 **Mitter C, Prayer D, Brugger PC, Weber M, Kasprian G.** 2015. In Vivo Tractography of Fetal Association.
517 *PLoS ONE*, 10(3):e0119536. DOI: 10.1371/journal.pone.0119536. PMID: 25742520.

518 **Newe A, Becker L, Schenk A.** 2014. Application and Evaluation of Interactive 3D PDF for Presenting and
519 Sharing Planning Results for Liver Surgery in Clinical Routine. *PLoS ONE*, 9(12):e115697. DOI:
520 10.1371/journal.pone.0115697. PMID: 25551375.

521 **Newe A, Ganslandt T.** 2013. Simplified Generation of Biomedical 3D Surface Model Data for Embedding
522 Into 3D Portable Document Format (PDF) Files for Publication and Education. *PLoS ONE*, 8(11):e79004.
523 DOI: 10.1371/journal.pone.0079004. PMID: 24260144.

524 **Newe A.** 2015. Towards an easier creation of three-dimensional data for embedding into scholarly 3D
525 PDF (Portable Document Format) files. *PeerJ*, 3:e794. DOI: 10.7717/peerj.794. PMID: 25780759.

526 **Phelps A, Naeger DM, Marcovici P.** 2012. Embedding 3D radiology models in portable document
527 format. *American Journal of Roentgenology*, 199(6):1342-1344. DOI: 10.2214/AJR.12.8716. PMID:
528 23169728.

529 **Qin Y, Yao W, Vu TT, Li S, Niu Z, Ban Y.** 2015. Characterizing Radiometric Attributes of Point Cloud Using
530 a Normalized Reflective Factor Derived From Small Footprint LiDAR Waveform. *IEEE Journal of Selected
531 Topics in Applied Earth Observations and Remote Sensing*, 8(2):740-749. DOI:
532 10.1109/JSTARS.2014.2354014.

533 **Ritter F, Boskamp T, Homeyer A, Laue H, Schwier M, Link F, Peitgen HO.** 2011. Medical image analysis.
534 *IEEE Pulse*, 2(6):60-70. DOI: 10.1109/MPUL.2011.942929.

535   **Ruthensteiner B, Heß M.** 2008. Embedding 3D Models of Biological Specimens in PDF Publications.
536   *Microscopy Research and Technique*, 71(11):778-786. DOI: 10.1002/jemt.20618. PMID: 18785246.

537   **Thoma GR, Ford G, Antani S, Demner-Fushman D, Chung M, Simpson M.** 2010. Interactive Publication:
538   The document as a research tool. *Web Semantics*, 8(2-3):145-150. DOI: 10.1016/j.websem.2010.04.001.
539   PMID: 20657757.

540   **Tory M, Möller T.** 2004. Human Factors in Visualization Research. *IEEE Transactions on Visualization and*
541   *Computer Graphics*, 10(1):72-84. DOI: 10.1109/TVCG.2004.1260759. PMID: 15382699.

542   **van de Kamp T, dos Santos Rolo T, Vagovic P, Baumbach T, Riedel A.** 2014. Three-Dimensional
543   Reconstructions Come to Life – Interactive 3D PDF Animations in Functional Morphology. *PLoS ONE*,
544   9(7):e102355. DOI: 10.1371/journal.pone.0102355. PMID: 25029366.

545   **Zlatanova S, Verbree E. 2005.** The third dimension in LBS: the steps to go. In *The 3rd symposium on*
546   *LBS&TeleCartography, 28-30 November. In: Geowissenschaftliche Mitteilungen, Heft Nr. 74*; Vienna,
547   Austria. 185-190.

548

549   *Supplemental Files*

550

551   **File S1. Example Figure.**
552   This PDF is an example which has been created with the described software.

553

554   **File S2. Software Manual.**
555   This PDF describes how to use the software. A HTML version of this documentation is also
556   included in the installers and can be viewed directly inside MeVisLab.

557