Interactive interpretation of 3D surfaces in field-based geosciences using mobile devices - concepts, challenges and applications

Melanie Kröhnert[a,*], Christian Kehl[e], Sophie Viseur[b], Simon J. Buckley[c,d]

[a] *Institute for Photogrammetry & Remote Sensing, TU Dresden, Helmholtzstr. 10, 01069 Dresden, Germany*
[b] *Aix Marseille Université, CNRS, IRD, CEREGE UM 34, Dept. Sedimentary and Reservoir Systems, 13001 Marseille, France*
[c] *Uni Research AS – CIPR, Nygårdsgaten 112, 5008 Bergen, Norway*
[d] *Department of Earth Science, University of Bergen, Allégaten 41, 5007 Bergen, Norway*
[e] *Danmarks Tekniske Universitet, DTU Compute, Richard Petersens Plads, Building 321/208, 2800 Kongens Lyngby, Denmark*

Abstract

Keywords: discrete geometry, surface reconstruction, volume reconstruction, surface parameterization, digital outcrops

2010 MSC: 00-01, 99-00

1. Introduction

A considerable number of domains within the geosciences rely on digitised natural observations and their interpretation to steer and constrain numerical models. Published (semi-)automatic interpretation methods emerged within the past decade that support the digital documentation of observations and interpretations. These advanced interpretation techniques require increasingly complex computing that is restricted to office-based work environments, which poses a problem for field-based studies. Domains such as hydrology, geology or glaciology hence established multi-stage procedures where observations are taken manually in the field and later digitised in the office. This is disadvantageous

*Corresponding author

Email addresses: `melanie.kroehnert@tu-dresden.de` (Melanie Kröhnert), `chke@dtu.dk` (Christian Kehl), `viseur@cerege.fr` (Sophie Viseur), `Simon.Buckley@uni.no` (Simon J. Buckley)

and within the referred domains, there is an increasing desire to facilitate digital interpretations in the field at the study location. Mobile computing equipment (e.g. smartphones and tablets) are one technological option to facilitate such digital field-based workflows. These devices are nowadays ubiquitous and can easily be equipped in field-based research. Also, as seen in technical magazines and the general media, the range of available devices continuously increases, which allows to find a "fit-for-purpose" device to each specific situation. New application cases, which are demonstrated and discussed in this article, and commitment within geoscience- and computer technology industry lead to an increasing interest in this cross-disciplinary domain between mobile computing and geoscientific interpretation.

Next to easily available, pocket-format computing devices, the required three-dimensional base data for modern applications also need to be available and being processed in a "mobile-ready" manner. The availability of topographic 3D surface data is steadily increasing due to easy-to-use software and instrumentation for surface generation (e.g. drones, structure from motion (SfM) [**?**] and multi-view geometry [**?**], satellite digital elevation models (DEMs)). Furthermore, crowdsourced data and Volunteered Geographic Information (VGI) contribute to the geoscience data inventory, which is acquired by citizen scientists.

Domain-specific mobile software is required in order allow for data interaction on the available mobile devices. Specific challenges such as power consumption, multi-manufacturer support, smart sensor utilisation and device intercommunication distinguish mobile software from common desktop software. This leads to a very different design electronics design of tablets and smartphones in comparison to desktop PCs and laptops, which in turn means that existing approaches for digital data processing and interpretation are not transferable as–is to this new computing domain. Even when considering the fast technological development, there are some challenges within mobile device software development that are rooted in the technology itself: user interfaces need to be specifically designed for mobile devices so to utilise touch screen interfaces, natural language interfaces and gesture interaction (e.g. ”swipe“ and ”optical lens“ motions). global navigation satellite system (GNSS)-based geo-localisation accuracy, as delivered by the integrated-circuit sensor of mobile, is inferior to common user expectations and requirements in geoscientific studies. The modalities of sensor data delivery (be it hardware sensor or software emulation), photo capturing and processing, and the computational capabilities of mobile devices differ significantly between each vendors. Short-comings such as inappropriate data structuring, visual object correlation and registration, increasing data volumes and the unavailability of off-the-shelf program codes further complicate the technological development. Addressing the demonstrated challenges distinguishes the mobile application development and common desktop software development for geoscience purposes. the same as in line 34?

This article demonstrates how the above-listed challenges can be addressed to provide, in the end, the desired added value for field-based research. This demonstration addresses the 3D data annotation and interpretation for two use cases within the domains of surface hydrology and (petroleum) geology. The content covered in the article is a detail-driven extension of earlier published research [1], focussing on extensive measurements to verify the reasoning and statements of previous studies.

The sections within this article adhere to the following structure: First, the use cases are presented as opening statements to introduce field-related tasks that are to be addressed with mobile device technology. Secondly, different 3D surface data representations are introduced that employed within this technical research. Thirdly, algorithmic baseline concepts that are key for interpreting 3D data on mobile devices are introduced, summarising project-internal development by the authors as well as referencing key literature on the subject. Fourth, the algorithms are mapped to the specific mobile technologies and components. The technologies and major parameters that impact the target use case application are highlighted. Finally, we showcase and discuss how available mobile systems are used in application scenarios from hydrology and petroleum geology to improve data analysis and integrate outdoor measurements in digital workflows. Then, the article is finalized with some concluding remarks and a discussion for future developments in this research trajectory.

2. Target case studies

TO-BE-FILLED

3. Representation basis – Geometry and Radiometry

Various representation forms for 3D terrain data are available. While early digital systems used DEMs for their simplicity and compact storage [2, 3], digital surface models (DSMs) and triangulated irregular networks (TINs) are dominating most terrain-based systems for application-specific analysis. An useful example can be seen a [4] for glaciology, where the authors use a triangulated digital surface model to represent a Patagonian glacier front. The same application of ... can be observed for climate studies, as in ... . In this context, it is important to distinguish geometrically valid TINs from polygon soup surfaces. While the latter is often employed in early stages of mesh-based software systems due to its simplicity and ease of implementation, valid TINs are employed in mature stages of the analysis. This is because some automated analysis (e.g. auto-interpretation, volume derivation) require clean surfaces with coherently outward-oriented surface normals.

In geoscience domains such as petroleum geology, texture- and color information are vital for interpretation- and analysis tasks. In these cases, as demonstrated by Buckley et. al [5] and Caumon et. al [6], the TIN is supplemented with photographic information that is projected on the surface as textures. As it is also possible to project textures from outside the visible spectrum as supplementary information on the surface [**?**], we generalize for the remainder of the article that the model in question consists of its geometric and radiometric information.

[EXAMPLE PHOTO FROM LIME]

In other geoscience domains, such as hydrology and free surface flow management, on the one hand georeferenced laser scanner point clouds, and on the other one colored point data streams provided by terrestrial photogrammetry for small- or unmanned aerial vehicle (UAV) for large-scale use cases, both processed by application of SfM, are used for several tasks like coastal monitoring [7, 8], soil erosion and rain-induced landslide observation or even monitoring river's topography [9]. Nevertheless, new approaches for low-cost and on-the-fly river monitoring [10] and simulation [11] arise due to globally increasing flash flood events after heavy rainfalls [12] that are further addressed in section. Since SfM became state of the art in geosciences, the acquisition of (true-)colored "point cloud" models is not that difficult and commonly employed because of their rapid processing (compared to conventional approaches like terrestrial laser scanning (TLS)). Regarding 3D annotation, nearest neighbour analysis provide an opportunity whereby surface triangulation can be avoided.

The stated base concepts of geometric and radiometric model information and their representations are also valid for mobile device software. Because of the limited processing speed of mobile chipsets, a tendency for employing point cloud models on mobile devices (e.g. Garcia et. al [13]) can be observed. A drawback of point cloud models is their sparse nature, which prohibits continuous analysis of field data. Therefore, DEMs have seen a revival in the mobile computing domain because they provide dense, closed geometric models that can be rendered and processed very effectively. Furthermore, with the inferior memory capacity of mobile devices in comparison to laptops and workstations, it is advantageous to be able to aggressively compress the data for mobile use, which is facilitated both by point clouds and DEMs. Base mapping applications such as Open Street Maps and Google Maps use ... as their main 3D data representation. Other systems within the geosciences processing 3D data on mobile devices, such as "Outcrop" and Geological Registration and Interpretation Toolset (GRIT), employ genuine textured triangulated DSM.

The form of model representation chosen significantly impacts the algorithms and analytical capabilities employed on the mobile device. Although all algorithms presented in this article work on either form of representation, some of the algorithms favor the treatment of triangulated surfaces (e.g. image-to-geometry registration, guided interpretation), while others clearly favour point-based representations (e.g. SfM and rendering).

4. Algorithms

This section demonstrates novel- as well as existing algorithms and methods on mobile devices that are used for solving analysis tasks in the geoscience use cases laid out in section **??**. As mentioned before, the effectiveness of each algorithm depends on the applied model representation.

4.1. Image-to-geometry registration

- ~~short description what it does~~
- ~~how is it generally employed~~
- ~~relation to mobile devices~~
- ~~approaches and implementation known on mobile devices~~
- integration to the above-mentioned use cases

Image-to-geometry algorithms aim at registering a 2D image to a given 3D surface, providing a transformation from 2D coordinate system to 3D coordinate system as follows:

$$P' = \begin{pmatrix} u \\ v \\ w \end{pmatrix} = [R_{3,3}|T_{1,3}] \cdot P \tag{1}$$

$$P = \begin{pmatrix} x \\ y \\ z \end{pmatrix} \tag{2}$$

$$P' \in \mathbb{R}^2 = \frac{P'}{w} \tag{3}$$

Using this coordinate system transformation on combination with a known interior camera orientation, it is possible to projectively map entire images on the surface just as surface texturing does. It is also possible to map specific objects on the image, such as image-based interpretations, on the surface. In the geosciences, these algorithms are employed to create a direct correlation between 3D model and the screen- or image space on which annotations and interpretations are based on [14]. The technique is also employed to provide texture mappings for multi-layer images, such as hyperspectral data or domain-specific thematic maps.

- feature-based registration: detect prominent points or edges in the input 2D images and a rendered representation of the 3D model
- different concept for registration:
  - for mesh models, 2D feature locations are raycasted using camera projection and the 3D model in background; 3D feature location determined upon ray-plane intersection
  - for point-based models, raycasting doesn't directly work as point cannot be intersected; prominent solution is employing a smart rendering technique that expands the point into an area, and generate a depth map; after depth map generation, the 3D coordinate of a 2D feature map can be read directly from the depth map; drawback of the method / offset for speed: accuracy limitation of depth maps; high-resolution depthmaps (above 512^2) cost a lot of performance
- registration then has source- and target 2D-3D point pairs (in a normalized manner) that are put into a least-squares optimization system (give math here)
- optimization system usually non-linear, usually employing Levenberg-Marquardt optimization schemes [15]

- largest challenges for feature-based registration: (a) reliable feature correlation and (b) optimization stability; useful constraints can employed in both cases – such as horizon information, building edges, object outlines – to increase the registration reliability and accuracy
- employing constraints is highly application-dependent
- feature-based registration techniques most prevalent on mobile devices due to execution speed, implementation ease and so forth
- examples of mobile implementation: [16**?** , 17**?**]
- drawback of feature-based methods: reliability and instability to imaging variances (discussed later in this article)
- a contrasting technique commonly achieving more accurate results: mutual information
- idea: pixel-wise comparison between 2D input image and 2D rendering of 3D scene; if both images match (meaning: $argmin\delta(I_{2D}, I'_{3D})$), then registration is completed
- mutual information [18] use notion of self-information and entropy to measure $\delta(I_{2D}, I'_{3D})$
- challenge: optimization of 7 degree-of-freedom system over such overdetermined system (i.e. each image pixel results in 2 equations for the optimization scheme) unstable and prone to local minima
- only optimization known to provide stable results is NEWUOA (i.e. Powell's method) [19];
- has recently been employed in an earth science case for light detection and range (lidar) registration [**?**]
- not available for mobile devices until now

- in terms of usability on mobile devices (why should we want to have that on mobile devices), projection of image-based information is vital on mobile devices
- because of interaction difficulty, image-based interpretations on 3D surface data is preferable
- it is potentially advantageous, in terms of power consumption, to implement data interaction in 2D rather than 3D; a hypothesis proven by experiment in this article
- as it directly relates to the interactive component of the geoscience app, it has to be executed in real-time on the device instead of an offline process

4.2. Data presentation and rendering

- ~~short description what it does~~
- ~~how is it generally employed~~
- ~~relation to mobile devices~~
- ~~approaches and implementation known on mobile devices~~
- integration to the above-mentioned use cases
- rendering the 3D model in this context refers to generation of image generation of 3D model by projective rasterization of model data to 2D imaging plane of a virtual camera
- rendering is performed to view the 3D model on the mobile device
- also employed to generate a reference image used in the image-to-geometry registration and the annotation and interpretation of the data
- major concepts: mesh-based rendering and Point-based Rendering (PBR)
- technical details on how this is done covered in the technology section **??**

4.2.1. Mesh-based rendering

- well-known concept
- mesh is transferred as vertex set and primitive set (i.e. triangles or polygons, depending on employing TIN or polygonal soup geometry) to graphics processor
- virtual camera is set up as projective view matrix
- primitives are positions in 3D scene via model transformation matrix
- rasterizer projects information to camera plane and transforms continuous data into discrete pixel-based representation
- in-time decompression can be optionally employed (see [20])
- texturing is employed to map image information on the surface; correlation between surface patches and texture images established explicitly beforehand or implicitly defined by geographic coordinate systems (e.g. WGS84 or UTM) for georeferenced surface

4.3. A novel approach to mobile point-based rendering

In comparison to mesh-based rendering, simple point projection seems to be a nice alternative saving computational resources and efforts for post-processing concerning outlier removal due to falsely surface reconstruction (e.g. blobs due to critic point normals). Thus, we simply project object points onto image plane using perspetive projection with assumption of distortion-free ideal camera with centred principle point. Thus, the camera matrix **K** equals identity matrix **I** and can be neglected in the following equations that generally base on notations given by Szeliski(2010). First, applying a six-parameter transformation transfers three-dimensional object points from world reference frame $\vec{X}_W$ into a 3D camera system $\vec{X}_c$ using

$$\vec{X}_c = \mathbf{R}\left(\vec{X}_W - \vec{X}_0\right) \tag{4}$$

where $\mathbf{R}$ is a $3x3$ orthonormal rotation matrix and $\vec{X}_0$ the translation vector to camera's projection center. For simplicity, the usage of the planar Cartesian UTM system with x pointing to the east and y pointing to the north with respect to the prevalent zone number. For z component, the height over the Earth Gravitational Model 1996 (EGM96) is advisable to use. Counting for homogeneous coordinates, we can describe the relation between camera $\vec{X}_c$ and image coordinates $\tilde{x}$ involving their depth components.

$$\begin{pmatrix} \tilde{u} \\ \tilde{v} \\ c_c \end{pmatrix} = \begin{pmatrix} x_c \\ y_c \\ z_c \end{pmatrix} \tag{5}$$

For image plane we introduce camera constant c_c that defines the distance between camera's sensor and projection center in $[mm]$ and is also known as focal length f. To separate camera sensor system and image system, we use the term c_c when talking about sensor $[mm]$ and f for digital image coordinates $[px]$. For conversion, c_c must be divided by sensor's pixel pitch. For 3D to 2D projection, homogeneous coordinates must be divided by their depth components resulting in inhomogeneous coordinates.

$$\vec{X}_{Cam} = \begin{pmatrix} \frac{\tilde{u}}{c_c} \\ \frac{\tilde{v}}{c_c} \\ 1 \end{pmatrix} = \begin{pmatrix} \frac{x_c}{z_c} \\ \frac{y_c}{z_c} \\ 1 \end{pmatrix} \tag{6}$$

Thus, two-dimensional coordinates can be described with

$$\begin{pmatrix} \tilde{u} \\ \tilde{v} \end{pmatrix} = \begin{pmatrix} \frac{x_c}{z_c} \cdot c_c \\ \frac{y_c}{z_c} \cdot c_c \end{pmatrix} \tag{7}$$

For a final transformation of 2D sensor coordinates into image pixels, we must shift the origin to left upper corner and scale the coordinates that are still in global units by pixel's relation in meters per pixel p_s. Finally, we derive image coordinates (u, v) for an ideal camera using

$$\begin{pmatrix} u \\ v \end{pmatrix} = \frac{1}{p_s} \begin{pmatrix} \frac{x_c}{z_c} \cdot c_c - u_0 \\ \frac{y_c}{z_c} \cdot c_c - v_0 \end{pmatrix} \tag{8}$$

4.3.1. Calculation of 3D bounding box of interest and image plane

Referring to the described use case of situation-based water level determination using a smartphone-camera based gauge (6.1), we need to define a region of interest regarding 3D point projection to render only user's field of view (figure 1). Thus, bounding box defining points to be projected must be calculated using camera position and orientation from fused smartphone sensors. Thereby it must be noted, that the heading is used for viewing direction only, tilt and roll are excluded. Because of uncertainties regarding exterior information (**??**), bounding box must be expanded to cover more object space than described by sensors as well as cameras field of view. Because of possible noise due to positioning, constants r and dh describe the domain of projection center's uncertainties parallel to image plane. For errors in depth, we define the correction $c = \frac{r}{tan(H)}$ for shifting the projection center along camera axis. The

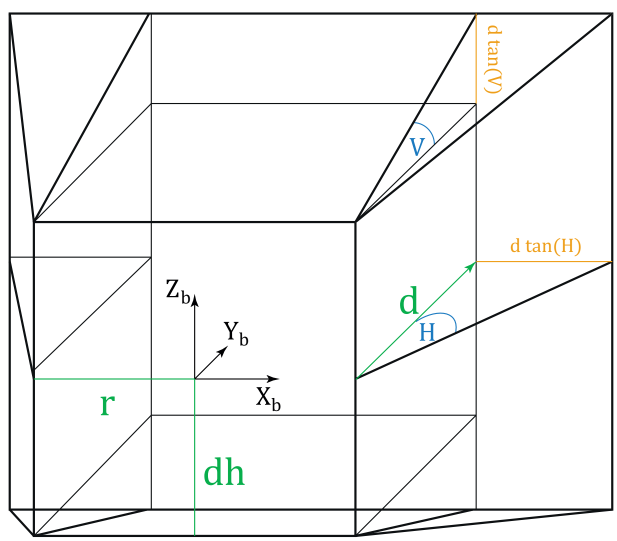

Figure 1: Bounding box definition.

box is widened by the horizontal H and vertical V opening angles with a fixed depth d. In order to generate reference data for image-2-geometry intersection to annotate 3D data by mobile imagery, the lateral accuracy given by mobile positioning system as well as the prevalent camera characteristics solve for the mentioned parameters. For camera based gauging, we set $d = 200[m]$. Regarding 3D point projection, each potential point will be checked laying in the box. Therefore, additional tiling of the 3D data set is advisable. Using the defined frustum of a pyramid as region of interest with local reference system, the image plane for 3D point rendering can be defined by perspective projection of the remote xz plane (1) with

$$\vec{X}_b = \begin{pmatrix} -r - d\tan H \\ d \\ dh + d\tan V \end{pmatrix}$$

for bounding box' background plane upper left and

$$\vec{X}_b = \begin{pmatrix} r + d\tan H \\ d \\ -dh - d\tan V \end{pmatrix}$$

lower right corner. Its height equals the height component in world reference frame z_w. Because of pyramid frustum, we finally must eliminate outer points between the smaller front and larger rear plane considering the eight extremes. workflow for outer point removal necessary?

4.3.2. Pyramid approach for depth filtering

Because of a limited range of pixels with defined size inside a image plane it seems to be obvious that in most cases more than one 3D object points corresponds to the same image pixel. Due to inhomogeneous coordinates it is not possible to figure out afterwards which points are in foreground compared to the camera distances and which ones are behind and so not visible. This problem can easily be solved during point cloud projection described above by a simple camera-to-object distance check. However, one problem still remains in case of e.g. glass fronts with lacking information (in TLS due to deflected lidar or SfM when having homogeneous surfaces) or small archs (see figure **??**) Then, points might be visible pointing away from camera projection center. On the one hand, point normals may solve the problem but due to data acquisition technique and model's complexity, they are more or less easy to derive (Sattler zitieren). Remedying image pyramids are a nice alternative approach used in this case. Therefore, multiple synthetic images are generated with step-by-step adjustment of p_s (see eq. 8), commonly by doubling which resulting in halve numbers of image rows and columns per layer. Than, the algorithm verifies if two pixels corresponds in two subsequent layers, preserving edges (figure 3, 2).

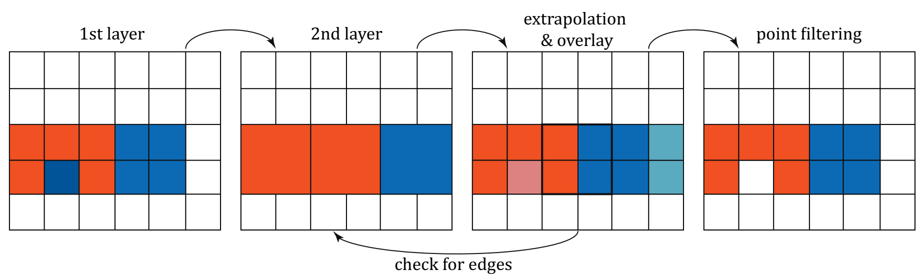

Figure 2: Visualisation of hierarchical depth filtering to handle point occlusions.

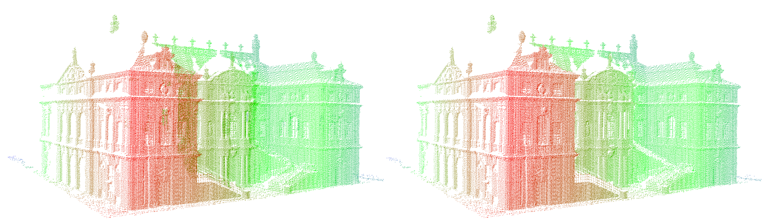

Figure 3: Left, actually obscured visible 3D points close to archs and windows. Right, edge preserving result after filtering.

4.3.3. Filling gaps due to missing points

TO BE FILLED BY MELANIE

4.4. Interpretation and annotation

- short description what it does
- how is it generally employed

- relation to mobile devices
- approaches and implementation known on mobile devices
- integration to the above-mentioned use cases

5. Technology

5.1. Sensors

5.1.1. Localization

- references: ...

5.1.2. Orientation

- stability IMU (see 3D-NO)
- precision IMU

5.1.3. Parameter sensitivity

5.2. Graphics

- as explained above, rendering 3D data is a key part for conducting annotations on 3D data, either based on 3D geometry itself or correlated images in camera space
- for mobile devices, this can technologically be realized two-fold, depending on the usage constraints
- generally, rendering stages (image-place projection, rasterization, tesselation & lighting, see [21]) can be realised by means of software or by hardware support
- here, compute operations (gaussian smoothing, derivative computations, vertex projection) are implemented on the mobile device CPU by available operations and libraries using the Android SDK (Java) and Android NDK (C++) on Google's Android system and Swift and Objective-C on Apple's iOS platform

- software-based rendering is more flexible in how operations are carried out as they do not need to account for hardware-specific processing pipelines; in the realm of mobile devices, non-standard software-based rendering operations are supported by a wider range of devices

- drawback of software-based rendering is performance as it makes suboptimal use newer computing capabilities and graphics-specific chipset operations

- example is the novel rendering approach for point-based rendering introduced in section **??**, which requires implementation flexibility

- most 3D rendering is done, even on mobile devices, with hardware-based rendering to varying degrees

- even when implementing specific rasterization algorithms (see section **??**), operations such as vertex projections and tesselation & lighting are performed on the GPU

- hardware-based rendering on mobile devices is facilitated by Khronos' graphics library for embedded systems (GLES) [22] on specialised mobile device graphics chips (e.g. Qualcomm Adreno, ARM Mali, NVIDIA Tegra)

- in comparison to software-based rendering, hardware acceleration provides improved performance

- drawbacks of hardware implementations are considerably longer development cycles, because the programming principles of GPUs differ considerably from common "App programming, and reduced flexibility on what can be realised

- for geoscientists and domain experts in the field, the distinction is good to be aware of as accelerated graphical apps for solving domain-specific tasks are usually offered as paid services to offset the increased development costs

- external circumstances and project constraints may govern implementation details for graphical systems
- major constraint imposed in this context is internet availability
- applications that are expected to operate in an urban setting or in well-developed infrastructure environments can make use of internet access
- this allows the externalisation of rendering tasks for 3D models and data to a network-oriented client-server architecture as in Ponchio et al. [20]
- field experts that use apps and require 3D-rendered information can submit rendering requests to a remote server that takes over the image generation of the 3D data
- technically, the app then only transmits metadata about storage location of the 3D data and receives the final, rasterised image, which allow energy-efficient operability
- furthermore, the process is agnostic to the specific mobile device generating the request, so this way of implementation works in the exact same manner for all mobile device regardless of the system manufacturer (e.g. Google, Apple, Microsoft)
- the reduced process load by externalising the rendering tasks allows using advanced algorithms for sensor tracking [] for improved localisation and orientation or augmented reality [] for information overlays and multimedia content
- in other geoscientific settings, such as field geology [] and environmental monitoring [] of remote areas, internet access is either restricted or expensive to establish
- subscription to satellite network with the required data rate costs around 70 euro per month $ (see `www.skydsl.eu`, skyDSL2+ flatrate with 30

MBit/s download); data rate needed as high-resolution images with real-time rendering rates (uncompressed 1920x1080 pixels resolution at 30 frames/s amounts to 1.39 GBit/s) requires highest data rates even when employing advanced compression techniques

- thus, for geoscience applications that operate in remote areas, web-based rendering is not an option
- in these cases, rendering needs to be done on the device
- for on-device rendering, the 3D data need to reside in the device memory and image generation needs to be done with performance-restricted hardware
- as is shown in section **??**, on-device rendering has considerable impact on the energy consumption
- on-device rendering that processes realistic field data requires considerable implementation efforts, as demonstrated by apps such as OpenWaterLevel [], GRIT [23] and Outcrop [24]
- principle scientific advances in mobile device rendering have demonstrated considerable progress over the years [13, 15, 25]
- scaling up smaller laboratory results with mobile graphics to realistics geoscience data, in terms of image quality and resolution as well as 3D base data, is a persisting challenge
- although technical development continuously provides more powerful devices, mobile devices need to sacrifice capabilities such as sensor availability as well as physical size and weight in order to provide larger memory space and higher-performance processors; examples for this trade-off manufacturing can be seen in special-purpose and high-performance tablets

such as NVIDIA Shield[1], Project Tango resp. ARCore[2] and Google Pixel C[3]

- an specific problem for geoscientists and domain experts in on-device rendering settings: trade-off between app response time, image quality, hardware utilization and cross-device operability
- in interviews amongst field geologists at the department of earth science at university of Bergen, a major demand from the target user base of such mobile app is interoperability between Android, Microsoft and Apple devices; demand originates from platform-agnostic working of common geoscience software on desktop computers for Apple and Windows
- on the other hand, app responsiveness and high image quality are amongst the next common priorities behind interoperability; user base asks for improved quality when operating advanced equipment (e.g. special-pirpose tablets, novel- and high-performance tablets)
- both demands are conflicting, as making use of specialised hardware (e.g. 64-bit, streaming SIMD extensions (SSE) and vectorisation, parallel processing and graphics processing unit (GPU) Computing such as CUDA[4] for image processing [26, 27], texture compression [28]) in turn means reducing the range of devices being able to operate the software
- these technologies are key as they provide technical solutions available right now to achieve the required responsiveness and image quality

5.3. Power consumption

- power consumption is a metric of major importance for mobile field applications

[1] NVIDIA Shield - `https://developer.nvidia.com/develop4shield`
[2] Google Augmented Reality - `https://developers.google.com/ar/`
[3] Google Pixel C- `https://www.android.com/tablets/pixel-c/`
[4] CUDA - `https://developer.nvidia.com/cuda-zone`

- metric governs the operation time of an app outdoors for specific studies; in application domains such as field geology, the target operation time is in the range of four to eight hours without recharging at an electricity plug
- for the use cases of waterline detection and field interpretation, we measured the energy consumption of the apps "OpenWaterLevel" and "GRIT" and its relation to technical indicators, such as central processing unit (CPU)- and GPU utilisation, memory consumption and environment measures that influence chip operations, namely the processor temperature
- the following measures for both apps were obtained on a Google Nexus 5 smartphone with 4-core ARM Cortex CPU and Qualcomm Adreno GPU; furthermore, OpenWaterLevels and its CPU-related measures were validated on a Samsung S8 smartphone with an 8-core ARM Cortex CPU of newer generation
- currently, the only app available on Android that allows measuring metrics on an app-specific level (i.e. logging the power consumption related to just one specific external app) is the Trepn Profiler [5]
- while this profiling app runs on all Android devices, the metrics that can be recorded (e.g. GPU load, processor temperature) vary between devices, so that GPU load measurements are not available for the Samsung S8 smartphone
- also the reason why measurements have been carried out on Google smartphones instead of other tablets; general insight on processor-power consumption behaviour are possible to be extrapolated to other devices
- in the first test, we compare the power consumption in relation to CPU- and GPU utilisation

[5] Trepn Profiler -

- our initial expectation is the a higher GPU load results in an increased power consumption compared to CPU-dominated operations, because mobile GPUs draw more power than CPUs to realise the increased graphics performance

- the results are shown for GRIT during 6.5 minutes of operations in fig. **??** (split in GPU and CPU contribution) and for OpenWaterLevel during 3.5 minutes of operations in fig. **??**

- observable in both apps: a clear dependency with CPU load and power consumption; shows that apps enter a state of conservative energy consumption if being inactive

- furthermore, when comparing CPU-dependent and GPU-dependent energy graphs, we observe that peak energy consumption strongly relates to GPU activity

- for measurements in "GRIT", we also have to distinguish between two operation modes

- actions such as image-to-geometry registration [] and 3D outcrop viewing employ 3D data processing and GPU computations in a major scale, while the image-based interpretation of an outcrop uses 2D image operations within Android-optimized data structures

- fig. **??** and **??** depict the 2D use case, whereas fig. **??** and **??** show the energy consumption in 3D operations

- as clearly observable in fig. **??** in comparison to fig. **??**, the 3D operations result in a drastic energy cost, raising the average power consumption by XYZ mA

- in contrast to novice expectation, the CPU load also increases in a 3D data processing setting

- that is because the CPU of the mobile processor needs to deliver the geometric- and texture data to the GPU; also, the CPU needs to decompress the texture image files, resulting in a higher processing load
- conclusions from the measurements are manifold
- on the level of assessing the profiled apps "OpenWaterLevel" for hydrological studies and "GRIT" for field geology studies, we obtained benchmark measures for their power consumption
- for OpenWaterLevels, the app can be operated with an average of XYZ mA per minute (XYZ mA/h), allowing a theoretical operability on a Google Nexus 5 smartphone of XYZ hours
- for GRIT, we distinguish between 2D- and 3D operability
- in 2D operation mode, only conducting image-based interpretations, GRIT consumes XYZ mA per minute (XYZ) per hours, allow XYZ hours of operation
- on the other hand, if being operated consistently in 3D-mode, the same app consumes XYZ mA per minute (XYZ) per hours, allow only 0.something hours of operation
- more generally speaking, the study shows that it is important for geoscience users to be aware of the which data they are working with on the mobile device
- even though 3D may be readily available for a given study, it is not advisable to access them on the mobile device over stretched periods, as the drastically increased power consumption results in an operation time of XYZ hours with two external power packs in the field bag
- considering the CPU load behaviour in 3D-mode, we can also hypothesize about the positive impact of utilising hardware-specific operations such as GPU texture decompression on energy consumption: while using the GPU

requires generally more power, it is also more efficient in operations such as texture decompression, therefore potentially having a positive affect on the overall power consumption of mobile field apps

6. Applications and Requirements

Due to the increasing usability of mobile devices for in the field annotations, several use cases concerning geosciences has become apparent. In the following, two essential

6.1. Derivation of hydrological parameters: Water level gauging

The last decade is characterized by a continued increase of globally devastating flash floods after heavy rainfalls. Even smallest creeks turned into hazardous streams causing flooding and landslides. Conventional gauging stations provide precise information about water levels measured over a short time period. State of the art techniques for administrative observation comprise water pressure sensors, floating gauges and conventional tide gauges. They are characterised by long-term stability and outdoor robustness providing accuries of several millimeters up to one centimeter [29]. Averaged over defined time intervals, it is advisable to remain caution regarding these accuracies may be too optimistic [30] .

Because of high costs in purchase and maintenance, gauging stations with complex sensing devices must be sparsely installed. A prime example here is the hydrological network in Saxony, Germany. Here, 184 gauging stations are installed for permanent observation on 154 of 259 rivers rising from small, medium and large catchments [31, 32]. Thus, around a third is not monitored neither during flood events when the most protection is required. Recently, commercial smartphone applications arose to enable crowd-sourcing based water level estimation for, among other things, such cases [33, 34]. But all of them have one thing in common: the water level is entered manually by engaged citizen scientists finding and photographing tide gauges close to rivers that makes - on the one hand a potential danger to themselves (f.e. by sudden landslides), and still limits on the other the approaches to open and visible gauges.

Improvements in this sense can be achieved through *image-2-geometry intersection* and 3D annotation for automatic water level determination without reference gauges for almost every situation regarding running waters. for this, the smartphone application *Open Water Level* that bases on the freely available open source camera framework *Open Camera* [35]. Open Water Level allows for free stationing water line detection using short hand held time-lapse image sequences (for details please refer to [**?**]). To interpret these, image measurements must be transformed into object space. Thus, exterior information needs to be provided by smartphone sensors for orientation and positioning.

6.1.1. Requirements applying to the sensors

To solve the task of autonomous water level determination on running rivers f.e. emergency cases using *image-2-geometry intersection*, citizen scientists position and orientation must be know. As figured out in 5, smartphone sensors accuracies for orientation and location are highly dependent on user's environment. Especially the strong correlation of heading and disturbing magnetic sources may be a issue must be solved specifically related to running rivers where metal railings usually exists. Similar effects can also be noted using high-end IMU systems for instance autonomous car navigation. But the magnetic influences inside cars are almost stable and can be calibrated during the drive (advanced navigation manual). For smartphone orientation, the magnetic strengths attaching the phone may change substantially in short time. A typical scenario would be: a citizen scientist walks along street, taking his phone inside the baggage close to metallic keys. While walking he passes several street lamps, signs, etc. Finally, he arrives at a bridge over a urban river, takes out the phone, looks down to the river and records the time lapse image sequence a few centimetres above a metallic railing. Meanwhile, several cars passing the same bridge. In this simple use case, the magnetic field around the smartphone changes countless times due to several unpredictable disturbances (table˙mag˙disturb) [36].

The heading angle has the highest influence compared to pitch and roll regarding 2D image and 3D object data registration. For this, a so-called synthetic image is rendered from colored 3D reference point clouds using scientist's location and orientation to define a situation-dependent bounding box of points to be projected onto image plane with respect to depth and indentations (see [37]). Thereby the heading defines the rotation of the depth direction, as a false angle gives a false viewing direction resulting in a synthetic image that has no similarity or only a little with the time lapse sequence. However, in case of no similarity and thus no possible solution for *image-2-geometry intersection*, simply no water level can be calculated. But in case of slight overlapping, there might be image matches but with very bad distribution that impedes a correct positioning (fig˙heading˙test) and may lead to even worse results of false water levels.

It is obvious that a second source for destructive results exists: the absolute geo-positioning using smartphones currently installed GNSS receivers. In urban scenes with several shadow effects due to high-rise buildings, errors of several meters in latitude and up to more than 30 meters in height are highly possible where even the weather has impact [38, 36, 39]. It is likely that, in the near future, smartphone's GNSS modules will be improved solving lateral accuracies of 50 centimetres [40].

For now, possible relief might come including other sources for positioning like digital elevation models for simple height correction or invoke map services that allows the user for position refinement. For this, some APIs are already provided by Google (quellen) but they are rather cost-expensive by extensive accessing. Another upcoming option is including barometers in sensor fusion, altitude can be measured within three meters [41] but for now, they are not a standard.

- (table, observation heading during water line detection outside → check magnetic strengthens and there changes over short times)
- (figure/table, sensitivity analysis → heading changed in terms of 10 degrees, what does it make for)

6.1.2. Requirements applying to the scenario

- *online processing and position refinement: need online connection*
- *image quality for water line detection: influence of image resolution, lightning, ...)*
- available approach to address the task

6.2. Field Geology

The goal of geological fieldtrips is to gather insight in the rock record and the structural- and sedimentary rock architecture of a given location. Rock architecture can be studied within subsurface seismic records, but this approach suffers from inferior imaging resolutions and physical limitations of the surveying technique. Therefore, surface outcrops are used for the study. Outcrops can be scanned with modern equipment (e.g. lidar [5, 42], drones [43] and SfM [44]) to generate digital surface representations. The most common representations of digital outcrops are coloured point clouds and textured TINs.

The geological aspect is introduced by interpreting the outcrop models. In this case, interpretations refer to (i) line marks for separating stratigraphic layers, (ii) surface-projected polygons to highlight structural- and sedimentary facies or specific architectural elements and (iii) minor ticks (e.g. lines, points, patterns) to indicate depositional attributes such as deposition orientation or grain geometry. The interpretations was until recently performed in a two-step process: sketches are drawn by hand in the field to document the field geologist's observation of the architecture. After the fieldtrip, the observations are digitalised in the office by transferring the sketched architecture on the available digital outcrop. From there on, further study goals (e.g. geomodelling) are pursued. As recently published, this workflow is currently being transformed into an integrated digital workflow in the field using mobile devices (see [45] for further details).

Geological interpretations can be documented on various scales, but from observations of the author most interpretations are conducted on medium-range.

This results in an average observation distance for architectural interpretations of between $100mto500m$ to document individual depositional elements and further distances of around $400mto1400m$ to document the overall stratigraphic framework of an outcrop. Therefore, as a result of perspective observations, the required lateral localisation accuracy is in the range of $\leq 2.5m$ for the individual element setting and $\leq 8m$ for the wide-angle stratigraphic setting. While achieving the former resolution can still be challenging with mobile sensors (see section **??**), the latter resolution is almost guaranteed for GPS localisation. The more important problem is in the vertical resolution: the vertical observation position has, especially in close-distance observations, a drastic impact on the view perspective. Even more important, a vertical localisation error of $\geq 1.5m$ may result in positioning the mobile device "under ground", making any image-based registration impossible. It is this vertical accuracy that is crucial for mobile device interpretation systems to work. Several improvements, such as DEMs and barometric altitude [46], have been proposed to reduce the vertical positioning error while there is still room for novel research proposals to provide more accurate vertical positioning or ground-based constraints on the altitude estimation.

One of the dominant challenges for digital field geology is the free availability of 3D surface models. Currently, research groups in the domain (e.g. from the University of Manchester, Durham University, University of Aberdeen, University of Bergen and UniResearch CIPR) are building their own digital outcrop databases. Due to the strong industry involvement, these and other databases (see SAFARI [47] and FAKTS [48]) are excluded from public access. Recent developments aim at providing digital outcrops in an open-access manner [49] to resolve the issue. Furthermore, due to the vertical positioning problem above, easy access to high- and medium resolution DEMs is important. As demonstrated by recent measurement, the usage of DEMs has a significant influence on the projection accuracy of image-based interpretation on mobile device towards 3D surface models [46].

One particular challenge in digital field geology is the treatment of environmental changes. Digital outcrops are infrequently collected and the textured models are used for field study all across the year. Therefore, in image registration terms, there is a drastic difference in local illumination, moisture content as well as fog and snow between acquired 3D surface models and the outcrop images collected during field trips. The issue has been previously discussed in terms of illumination differences [17], but drastic changes in terms of fog and moisture are still problematic to treat. Therefore, it is advisable to collect digital outcrop models for prominent locations in different seasonal conditions to allow for variety in model selection when planning field trips.

Currently available systems that provide digital outcrop interpretation capabilities on mobile devices in 3D include GRIT [23] and Outcrop [24], though earlier prototypes have been demonstrated [50]. Outcrop, developed by Centre Européen de Recherche et d'Enseignement des Géosciences de l'Environnement (CEREGE) at Aix-Marseille Université, is a mobile device app for Android devices that is able to load and process various forms of numerical outcrops. Its major focus is the documentation of structural features (e.g. fault areas, fractures and rock deformations) on outcrops using line interpretations. Furthmore, it is possible to pin extended note annotations to the model. GRIT, developed as a collaboration between UniResearch AS CIPR, University of Bergen, University of Aberdeen and CEREGE, is a mobile device app for Android devices that can handle large-area digital outcrops of tens of kilometres is surface length in 3D. Its major focus is the documentation of the sedimentary- and stratigraphic architecture (e.g. strata boundaries, depositional object envelopes, facies areas) on outcrops via lines, polygons and brushes. The interpretations are mapped in a 2D-3D interplay between outcrop surface and field photograph.

[comparison photo: GRIT and Outcrop]

- ~~recap: task to be solved~~
- ~~main requirements for (location- and orientation) sensor accuracy and~~ geometric accuracy
- ~~specific requirements to this use case: data availability; illumination;~~ net-

work inavailability

- ~~available approach to address the task~~

6.3. Virtual Field Trips

- recap: task to be solved
- main requirements for (location- and orientation) sensor accuracy and geometric accuracy
- specific requirements to this use case: data availability; illumination; network inavailability
- available approach to address the task

7. Conclusions

This article assessed the possibility of interactive interpretation and annotation of 3D outcrops on mobile devices in multiple geoscientific domains. Due to the research effort in recent years, novel mobile applications such as OpenWaterLevels for surface hydrology and GRIT for field geology were introduced to the community to bridge the gap between lab assessment and outdoor field work for data interpretation. This article also showed further application areas that build upon mobile device technology and the interactive annotation of 3D surface data for geoscientific problem solving.

McCaffery et al. proposed the use of mobile devices for field interpretation in geology in 2005 [3]. The technological specifics of mobile device app development hampered the progress on this goal for years. Only recent advancements in efficient treatment of 3D data [**?**], algorithmic proposals for image-to-geometry registration (see [16, 46]) and on-device 3D rendering (as presented in [25] and in this article for point-based surface) specifically designed for mobile devices make the actual use for geoscientific applications in the field possible. The utilisation of crowdsourced VGI and introduction of mobile devices as low-cost measuring devices for real-world problems [51] contribute to the acceptance of this mobile device technological development within the geoscientific community. Computer Vision challenges such as image registration under changing illumination conditions and with reduced image resolution can be viewed as "sufficiently solved" to make photogrammetric- and vision-based algorithms applicable to real-world outdoor settings, while still leaving space for improvement and quality and performance.

The measurements found in this article as well as its related studies suggest that localisation and orientation of mobile device sensors with respect to the application-specific accuracy requirements is a persisting challenge. The sensors employed by low-cost devices have accuracy limitations. Sensor filtering- and fusion techniques are required to even moderately consider the use of such sensor data. Environmental effects such as device-internal heating processes and the system-internal handling of sensor initialisation further complicate the calibration of such sensors without user involvement.

Furthermore, this study gives a representative overview about the energy consumption of mobile apps employing 3D surfaces, computer vision and computer graphics procedures. It was shown that the distinction between 2D- and 3D data used by mobile apps significantly drives the power consumption, and therefore the operation time of the mobile field apps during a study. Means of reducing the power consumption in the future have, next to extended periods of app use by domain experts, beneficial secondary effects: power-reduced main functions of the mobile app allow energy-expensive simultaneous localisation and mapping (SLAM) techniques to be used for sensor data augmentation.

Lastly, the treatment of vegetation within scanned- and photographed data during mobile field studies remains a challenge in the context of interactive interpretation. 3D reference data are obtained less frequent than they are used in a given outdoor setting. Vegetation itself is visually dynamic content that complicates image registration to existing 3D data, which complicates interpretations in common outdoor settings. While current procedures of data processing try to segment- and remove vegetation data from scans [], it leaves the mobile device app less information to work with when registering photos. Therefore, proposing means of 3D topographic data processing that homogenizes vegetation in 3D scans and photos without removing the related data will have an impact on accurate outdoor photo registration on 3D base data.

8. Discussion

- porting existing desktop algorithms on mobile devices [quick and fast]
- pre-processing of geodata for mobile use

Acknowledgements

Thank ESF, Richard Boerner for bounding box determination and image filtering

References

[1] M. Kröhnert, C. Kehl, H. Litschke, S. J. Buckley, Image-to-geometry registration on mobile devices - concepts, challenges and applications, in: L. Paul, G. Stanke, M. Pochanke (Eds.), 3D-NordOst, volume 20, Gesellschaft zur Förderung angewandter Informatik, 2017, pp. 99–108.

[2] I. Trinks, P. Clegg, K. McCaffrey, R. Jones, R. Hobbs, B. Holdsworth, N. Holliman, J. Imber, S. Waggott, R. Wilson, Mapping and analysing virtual outcrops, Visual Geosciences 10 (2005) 13–19.

[3] K. McCaffrey, R. Jones, R. Holdsworth, R. Wilson, P. Clegg, J. Imber, N. Holliman, I. Trinks, Unlocking the spatial dimension: digital technologies and the future of geoscience fieldwork, Journal of the Geological Society 162 (2005) 927–938.

[4] E. Schwalbe, H.-G. Maas, The determination of high-resolution spatio-temporal glacier motion fields from time-lapse sequences, Earth Surface Dynamics 5 (2017) 861–879.

[5] S. J. Buckley, J. A. Howell, H. D. Enge, T. H. Kurz, Terrestrial laser scanning in geology: data acquisition, processing and accuracy considerations, Journal of the Geological Society 165 (2008) 625–638.

[6] G. Caumon, G. Gray, C. Antoine, M. O. Titeux, Three-dimensional implicit stratigraphic model building from remote sensing data on tetrahedral meshes: Theory and application to a regional model of la popa basin, ne mexico, IEEE Transactions on Geoscience and Remote Sensing 51 (2013) 1613–1621.

[7] P. Letortu, M. Jaud, P. Grandjean, J. Ammann, S. Costa, O. Maquaire, R. Davidson, N. L. Dantec, C. Delacourt, Examining high-resolution survey methods for monitoring cliff erosion at an operational scale, GIScience & Remote Sensing (2017) 1–20.

[8] M. Medjkane, O. Maquaire, S. Costa, T. Roulland, P. Letortu, C. Fauchard, R. Antoine, R. Davidson, High-resolution monitoring of complex coastal morphology changes: cross-efficiency of SfM and TLS-based survey (vaches-noires cliffs, normandy, france), Landslides (2018).

[9] Y. Watanabe, Y. Kawahara, UAV photogrammetry for monitoring changes in river topography and vegetation, Procedia Engineering 154 (2016) 317–325.

[10] M. Kröhnert, R. Meichsner, Segmentation of environmental time lapse image sequences for the determination of shore lines captured by hand-held smartphone cameras, ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences IV-2/W4 (2017) 1–8.

[11] J. G. Leskens, C. Kehl, T. Tutenel, T. Kol, G. de Haan, G. Stelling, E. Eisemann, An interactive simulation and visualization tool for flood analysis usable for practitioners, Mitigation and Adaptation Strategies for Global Change (2015) 1–18.

[12] E. N. Mueller, A. Pfister, Increasing occurrence of high-intensity rainstorm events relevant for the generation of soil erosion in a temperate lowland region in central europe, Journal of Hydrology 411 (2011) 266 – 278.

[13] S. García, R. Pagés, D. Berjón, F. Morán, Textured splat-based point clouds for rendering in handheld devices, in: Proceedings of the 20th International Conference on 3D Web Technology, Web3D '15, ACM, New York, NY, USA, 2015, pp. 227–230. URL: `http://doi.acm.org/10.1145/2775292.2782779`.

[14] C. Kehl, S. Buckley, R. Gawthorpe, I. Viola, J. Howell, DIRECT IMAGE-TO-GEOMETRY REGISTRATION USING MOBILE SENSOR DATA, ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences 3 (2016) 121–128.

[15] C. Kehl, S. J. Buckley, J. A. Howell, Image-to-Geometry Registration on Mobile Devices - An Algorithmic Assessment, in: Proceedings of the 3D-NordOst workshop, volume 18, GFaI, 2015, pp. 17–26. ISBN 978-3-942709-14-9.

[16] S. Gauglitz, C. Sweeney, J. Ventura, M. Turk, T. Hollerer, Model estimation and selection towards unconstrained real-time tracking and mapping, Visualization and Computer Graphics, IEEE Transactions on 20 (2014) 825–838.

[17] C. Kehl, S. J. Buckley, S. Viseur, R. L. Gawthorpe, J. A. Howell, Automatic illumination-invariant image-to-geometry registration in outdoor environments, The Photogrammetric Record 32 (2017) 93–118.

[18] P. Viola, W. M. Wells, Alignment by maximization of mutual information, International journal of computer vision 24 (1997) 137–154.

[19] M. Corsini, M. Dellepiane, F. Ganovelli, R. Gherardi, A. Fusiello, R. Scopigno, Fully Automatic Registration of Image Sets on Approximate Geometry, International journal of computer vision 102 (2013) 91–111.

[20] F. Ponchio, M. Dellepiane, Multiresolution and fast decompression for optimal web-based rendering, Graphical Models 88 (2016) 1 – 11.

[21] J. Kessenich, G. Sellers, D. Shreiner, OpenGL Programming Guide: The Official Guide to Learning OpenGL, Version 4.5 with SPIR-V, OpenGL, Pearson Education, 2016. URL: https://books.google.fr/books?id=vUK1DAAAQBAJ.

[22] P. Mehta, Learn OpenGL ES: For Mobile Game and Graphics Development, Apress, 2013. URL: https://books.google.dk/books?id=RaO9AAAAQBAJ.

[23] C. Kehl, J. R. Mullins, S. J. Buckley, R. L. Gawthorpe, J. A. Howell, I. Viola, S. Viseur, Geological Registration and Interpretation Toolbox (GRIT): A Visual and Interactive Approach for Geological Interpretation in the Field, in: Proceedings of 2nd Virtual Geoscience Conference, 2016, pp. 59–60.

[24] S. Viseur, R. Roudaut, R. Bertozzi, M. Castelli, J.-L. Mari, 3d interactive geological interpretations on digital outcrops using a touch pad, in: Vertical Geology Conference (VGC), 2014.

[25] M. Agus, E. Gobbetti, F. Marton, G. Pintore, P.-P. Vázquez, Mobile Graphics, in: A. Bousseau, D. Gutierrez (Eds.), EuroGraphics 2017 - Tutorials, The Eurographics Association, 2017. doi:10.2312/egt.20171032.

[26] S. Heymann, K. Müller, A. Smolic, B. Froehlich, T. Wiegand, Sift implementation and optimization for general-purpose gpu, Winter School of Computer Graphics (WSCG) (2007).

[27] M. A. Hudelist, C. Cobârzan, K. Schoeffmann, Opencv performance measurements on mobile devices, in: Proceedings of International Conference on Multimedia Retrieval, ICMR '14, ACM, New York, NY, USA, 2014, pp. 479:479–479:482. URL: http://doi.acm.org/10.1145/2578726.2578798. doi:10.1145/2578726.2578798.

[28] D. Chait, Using ASTC Texture Compression for Game Assets, whitepaper, NVIDIA Corporation, 2015. URL: https://developer.nvidia.com/astc-texture-compression-for-game-assets.

[29] S. Siedschlag, Wasserstände und Durchflüsse - messen, speichern und übertragen im digitalen Zeitalter, in: Dresdner Wasserbauliche Mitteilungen, volume 53 of *Dresdner Wasserbauliche Mitteilungen*, Technische Universität Dresden, Institut für Wasserbau und technische Hydromechanik, 2015. URL: https://henry.baw.de/handle/20.500.11970/103357.

[30] I. Horner, B. Renard, J. L. Coz, F. Branger, H. McMillan, G. Pierrefeu, Impact of stage measurement errors on streamflow uncertainty, Water Resources Research (2018).

[31] Saxon Flood Centre, Water levels & flow rates, 2018. URL: https://www.umwelt.sachsen.de/umwelt/infosysteme/hwims/portal/web/wasserstand-uebersicht, accessed: 2018-03-05.

[32] U. Büttner, E. Wolf, Konzeption des gewässerkundlichen pegelnetzes in sachsen, 38. Dresdner Wasserbaukolloquium 2015 Messen und Überwachen im Wasserbau und am Gewässer (2015).

[33] S. Etter, B. Strobl, Crowdwater, 2018. URL: http://www.crowdwater.ch/de/home/, accessed: 2018-03-06.

[34] Kisters, Einfach smart: App für Pegelmessung auf Knopfdruck, 2014. URL: https://www.kisters.de/fileadmin/user_upload/Wasser/Produkte/WISKI/Produktblaetter/MobileWaterTracker_de_mail.pdf, accessed: 2018-03-06.

[35] M. Harman, Open Camera - Camera app for Android, 2017. URL: https://sourceforge.net/projects/opencamera/, version 1.38.

[36] J. R. Blum, D. G. Greencorn, J. R. Cooperstock, Smartphone sensor reliability for augmented reality applications, in: K. Zheng, M. Li, H. Jiang

(Eds.), Mobile and Ubiquitous Systems: Computing, Networking, and Services, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013, pp. 233 – 248.

[37] R. Boerner, M. Kröhnert, Brute force matching between camera shots and synthetic images from point clouds, volume XLI-B5, 2016, pp. 771–777. doi:doi:10.5194/isprs-archives-XLI-B5-771-2016.

[38] C. Bauer, On the (in-)accuracy of gps measures of smartphones: A study of running tracking applications, in: Proceedings of International Conference on Advances in Mobile Computing & Multimedia, MoMM '13, ACM, New York, NY, USA, 2013, pp. 335:335–335:341. URL: http://doi.acm.org/10.1145/2536853.2536893. doi:10.1145/2536853.2536893.

[39] P. A. Zandbergen, S. J. Barbeau, Positional accuracy of assisted GPS data from high-sensitivity GPS-enabled mobile phones, Journal of Navigation 64 (2011) 381–399.

[40] S. K. Moore, Superaccurate gps coming to smartphones in 2018, IEEE Spectrum (2017). Accessed: 2018-03-06.

[41] G. Liu, K. M. A. Hossain, M. Iwai, M. Ito, Y. Tobe, K. Sezaki, D. Matekenya, Beyond horizontal location context: measuring elevation using smartphone's barometer, in: Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing Adjunct Publication, ACM Press, 2014. doi:10.1145/2638728.2641670.

[42] S. J. Buckley, E. Schwarz, V. Terlaky, J. A. Howell, R. Arnott, Combining Aerial Photogrammetry and Terrestrial Lidar for Reservoir Analog Modeling, Photogrammetric Engineering & Remote Sensing 76 (2010) 953–963.

[43] T. J. Dewez, J. Leroux, S. Morelli, Uav sensing of coastal cliff topography for rock fall hazard applications, in: Journées Aléas Gravitaires JAG 2015, 2015.

[44] J. Chandler, S. Buckley, Structure from motion (SFM) photogrammetry vs terrestrial laser scanning, American Geosciences Institute (AGS), 2016.

[45] C. Kehl, J. R. Mullins, S. J. Buckley, J. A. Howell, R. L. Gawthorpe, Interpretation and mapping of geological features using mobile devices in outcrop geology - a case study of the saltwick formation, north yorkshire, uk, AGU Books - Special Issue (2018 (accepted for publication)).

[46] C. Kehl, S. J. Buckley, S. Viseur, R. L. Gawthorpe, J. R. Mullins, J. A. Howell, Mapping field photos to textured surface meshes directly on mobile devices, The Photogrammetric Record 32 (2017) 398–423.

[47] T. Dreyer, L.-M. Fält, T. Høy, R. Knarud, J.-L. Cuevas, et al., Sedimentary architecture of field analogues for reservoir information (SAFARI): a case study of the fluvial escanilla formation, spanish pyrenees, in: The Geological Modeling of Hydrocarbon Reservoirs and Outcrop Analogs, volume 15, International Association of Sedimentologists – Special Publications, Wiley Online Library, 1993, pp. 57–80.

[48] L. Colombera, F. Felletti, N. P. Mountney, W. D. McCaffrey, A database approach for constraining stochastic simulations of the sedimentary heterogeneity of fluvial reservoirs, AAPG bulletin 96 (2012) 2143–2166.

[49] A. J. Cawood, C. E. Bond, erock: an online, open-access repository of virtual outcrops and geological samples in 3d, in: EGU Geophysical Research Abstracts, volume 20, 2018, p. 18248.

[50] L. Hama, R. A. Ruddle, D. Paton, 3d mobile visualization techniques in field geology interpretation: Evaluation of modern tablet applications, in: AAPG Hedberg Research Conference: 3D Structural Geologic Interpretation: Earth, Mind and Machine, 2013.

[51] A. Eltner, H. Sardemann, M. Kröhnert, E. Schwalbe, Camera based low-cost system to monitor hydrological parameters in small catchments, in: EGU General Assembly Conference Abstracts, volume 19, 2017, p. 6698.

Highlights

- point 1
- point 2
- point 3
- point 4
- point 5