Streaming and 3D mapping of Agri-data on mobile devices

V. Stojanovic, R. Falconer, J. Isaacs, D. Blackwood,
D. Gilmour, D. Kiezebrink and J. Wilson

The published article is available from doi:
http://dx.doi.org/10.1016/j.compag.2017.03.019

Streaming and 3D Mapping of Agri-Data on Mobile Devices

V. Stojanovic[1], R. Falconer[1*], J. Isaacs[2], D. Blackwood[1], D. Gilmour[1], D. Kiezebrink[3] and J. Wilson[3]

[1] Abertay University, Arts, Media and Computer Games, Kydd Building, Bell Street, Dundee, DD1 1HG, Scotland, UK

[2] Robert Gordon University, Garthdee House, Garthdee Road, Aberdeen, AB10 7QB, Scotland, UK

[3] Soil Essentails Ltd, Hilton of Fern Farm, Brechin, DD9 6SB, Scotland, UK

Abstract

Farm monitoring and operations generate heterogeneous AGRI-data from a variety of different sources that have the potential to be delivered to users ‘on the go’ and in the field to inform farm decision making. A software framework capable of interfacing with existing web mapping services to deliver in-field farm data on commodity mobile hardware was developed and tested. This raised key research challenges related to: robustness of data steaming methods under typical farm connectivity scenarios, and mapping and 3D rendering of AGRI-data in an engaging and intuitive way. The presentation of AGRI-data in a 3D and interactive context was explored using different visualistaion techniques; currently the 2D presentation of AGRI- data is the dominant practice, despite the fact that mobile devices can now support sophisticated 3D graphics via programmable pipelines. The testing found that WebSockets were the most reliable streaming method for high resolution image/texture data. From our focus groups there was no single visualisation technique that was preferred demonstrating that a range of methods is a good way to satisfy a large user base. Improved 3D experience on mobile phones is set to revolutionize the multimedia market and a key challenge is identifying useful 3D visualization methods and navigation tools that support the exploration of data driven 3D interactive visualisation frameworks for AGRI-data.

Keywords: Interactive Visualisation; Farm Management Integrated Systems; Precision Agriculture; Data Aggregation; Mobile Devices; 3D Graphics

---

*Corresponding author at: School of Arts, Media & Computer Games, Abertay University, Kydd Building, Bell Street, DD1 1HG, Dundee, Scotland, UK.

Email: r.falconer@abertay.ac.uk

1. Introduction

Delivering secure and sustainable provision of food, water and energy, particularly in the face of climate change and reduced carbon targets is a huge challenge. Precision Agriculture (PA) and sustainable intensification has been advocated as a scalable solution to modern global food security challenges by saving time, energy, water and money (Karetsos and Sideridis, 2014; Whitacre and Griffin, 2014; Santana et al., 2007). PA stemmed from the desire to manage farms more sustainably. Traditionally PA has been restricted to those that can afford the latest technology, but maturation and ubiquity of enabling digital and mobile technologies are set to transform PA (Whitacre and Griffin, 2014; Karetsos and Sideridis, 2014; Butler 2006). This is supported by various UK, USA and EU strategies for encouraging innovation in agriculture (e.g. UK Agri-Tech Strategy (HM Government, 2013) and associated AGRIMETRICS (Tiffin, 2017) and EUs FIWARE (López-Riquelme *et al.*, 2016) accelerators) supporting a revolution in the use of data science from “farm to fork”.

Precision Agriculture (PA) is tightly coupled to the Internet of Things (IoT) and converting big data, originating from heterogeneous sources, into information is a key challenge (Mulla, 2013; Zhang et al., 2002). There is however a growing need for “on the go” decision-making tools for in-field viewing of relevant farm data (Ying, 2012; Chittaro, 2006; Pombinho et al., 2007). Mobile technology that interfaces with existing farm servers could deliver data that offers early warnings of potential issues in the field e.g. assessing the risks of disease and pest outbreaks or poor crop performance. The authors see such a mobile tool as complimenting the rich landscape of Farm Management Information System (FMIS) presented by Fountas et al., (2015) and illustrated in Fig 1. However, to progress there are two technical challenges that need to be addressed:

- Streaming data efficiently from a farm server to a commodity mobile device

- Implementing and evaluating different interactive 2D and 3D visualisation methods for the display of AGRI data on a mobile device

Previous mobile applications (*apps*) have been developed for farmers and agronomists, but these apps are focused on specific needs (e.g. soil nutrient approximation), and utilise 2D visualisation methods (Hopkins, 2013). Mobile devices (tablets and/or smart phones) are now ubiquitous with more memory, faster processors and feature a programmable Graphic Processing Unit (GPU) (Shebanow, 2013). GPUs can be programmed via special programs called *shaders*, which permit sophisticated mobile graphics once reserved for video games and PC-based visual simulations (Akenine-Möller et al., 2008; Falconer et al., 2015). Mobile graphics hardware is designed to work with texture data efficiently. The benefits of using high resolution aerial photography (Lange, 2001) and interactive 3D landscapes (Lovett et al., 2015) for enhancing user engagement has been highlighted. Additionally mobile GPS hardware can be exploited to ensure relevant data is delivered to users by linking GPS to the Field of View (FoV) (Burigat and Chittaro, 2015; Tsiropoulos & Fountas 2015).

Recently (PIX 4D, 2016; Puri, 2016) released software to construct 3D textured Digital Elevation Models (DEM) of FARM DATA, captured using unmanned aerial drone, or using sensors. There is a growing recognition in the AGRI sector that 3D visualisation is a useful tool as exemplified by Gepiel et al., (2015), where a PC-based 3D visualization of in-field sensor data is created. Areview of ICT-AGRI ERA-NET EU funded projects for 2010 to 2015 features few utilising 3D content with the exception of VAROS (Jordan 2015). Further, there is a paucity of mobile applications for PA with interactive 3D visualisation and this is primarily a consequence of two issues. Firstly, the skills set associated with 3D graphics does not intersect with the traditional AGRI sector. Secondly, the real benefits of mobile 3D content have yet to be discovered in this sector. At the time of writing, the authors were not able to find a specific example of a 3D visualization specifically for crop yield analysis on a mobile platform.

A software framework for streaming and rendering data in 3D, with potential applications to crop scouting, is presented based on mobile game technology. The software framework combines virtual texturing and streamed farm data to inform ‘on the go’ decision making. The technology is demonstrated using crop yield data and high resolution aerial photography although it can in principle display other AGRI data. The proposed AGRI-AG mobile app, enabled only by the multidisciplinary convergence of game technology with AGRI data, has the potential to transform in-field crop monitoring and inform early decision-making by growers to improve efficiency/profitability of the farming industry, providing healthier, more affordable food for the future.

2. Software Development

2.1 Application

The Model Viewer Controller (MVC) is a common and well documented software design pattern (Vlissides et al., 1994) and this methodology guided the development of the app.. The MVC pattern is widely used and suitable for applications that require user input via a graphical user interface (GUI). The MVC pattern is also the default and recommended software design pattern for developing Android applications (Phillips and Hardy, 2013).

Insert Figure 1 here

Figure 1 illustrates how the AGRI-AG app can integrate into the existing FMIS landscape, which is reviewed in Fountas *et al.*, (2015), to support crop monitoring illustrated here by delivering *yield maps.*

Insert Figure 2 here

Fig. 2 shows the components of the AGRI-AG application, implemented as an Android mobile app and highlighting the data streaming, processing and rendering stages.

AGRI-AG user input is facilitated through the mobile app's user interface as well as GPS functionality. Users can navigate the 3D scene using gestures for zooming, rotating and panning the 3D scene. The GPS coordinates are used to centre the users view in the 3D scene, which acts as a virtual camera so that users can freely navigate the scene. The different methods for AGRI-data presentation is by the toggling of radio buttons.

The 3D scene comprise a textured Digital Elevation Model (DEM) and different methods to present yield data (in 2D and 3D). Two streaming methods delivering large textures e.g. high resolution aerial photography from UAV, but this could also be satellite infrared imagery for assessing crop health, are investigated. Although the desire is to integrate the mobile technology with existing farm servers, for this research the test data (spatio-temporal yield data and high resolution imagery) was stored on a remote server located at the university, as the PA company is a live business operation.

2.2 Data Format Specifications

The main data types that AGRI-AG deals with are image (textures) and text files. The image files are used to generate the 3D geometry for the Digital Elevation Model, as well to provide texture overlays for the yield data and aerial photography. The image data files are JPG image files which are faster to decode on the mobile tablet hardware and have reasonable compression (Thiagarajan, 2012). The text files store yield data values that are parsed and used to generate representative 3D primitives. The text used to store the sampled yield data is stored as a standard CSV (Comma Separated Value) text file. The streamed data from the server is encoded as Base64 string data files. This is a convenient format that encodes the data to a Base64 hexadecimal ASCII file encoding. This format is used because it requires less calls to be made to the server and the required data is packaged and sent as a single Base64 data file form the server to the client, using either long-polling HTTP or WebSockets-based client/server communication model (Popov, 2009). Presently the yield maps used for the visualisation by AGRI-AG are not generated in real time. Instead they were generated offline using yield mapping software, GS+ (Gammadesign Software, 2016). Generating yield maps require the use of *Kriging* algorithms, which are compute intensive, but could be a prime candidate for parallelisation on mobile devices in the future. The CSV file is used i) as input into a Block Kriging algorithm to generate the interpolated yield map images mimicking what would be done on the farm server. These images are then exported, along with a standard colour table used by GS+, as JPEG image files, and transferred to the test server which can be downloaded by the app as needed. Figure 3 below illustrates the process of acquiring the yield data which is then presented in various forms as described below.

Insert Figure 3

2.3 The 3D Rendering Pipeline and Data Visualisation

The AGRI-AG app features various 2D and 3D visualisation methods based either on textures or 3D primitives, that represent the wider agricultural context and crop yield data. Since visualisation methods can be prohibitively expensive to compute on the CPU, the GPU is used to offload the required processing from the CPU. The visualisation methods used for AGRI-AG are implemented in a *shader* written in GLSL (OpenGL Shading Language, a C-like programming language for shaders (Munshi, 2008)) as part of the 3D graphics programmable pipeline. *Vertex shader* code is used to define how the GPU will handle the vertex data associated with the 3D objects (Brothaler, 2013). The vertex shader computes the vertex position, vertex normal and the texture coordinates of a 3D object being rendered. This data is streamed to the *fragment shader* which computes the final pixel colour based on the object colour, texture (image data) and shading model used. Basic Gouraud shading is implemented on a per-vertex basis, and is used to combine the texture, scene lighting and 3D object colour (Gouraud, 1971).

2.3.1 Texture-based Landscape Visualisation

The 3D Digital Elevation Model (DEM) that captures the topography of the landscape is represented using an image (Mach and Patschek, 2007). This image can either be taken by a UAV or obtained via third party sources (such as Ordnance Survey UK). Increasingly this type of image data is large in terms of resolution and must be resized and resampled before use on mobile devices. Using standard graphics programming approaches 2D textures, also represented as an image, can be mapped onto the 3D DEM. These 2D textures can be either high resolution aerial photography, capturing features of the landscape, or colour-mapped yield data derived from block Kriging algorithms. To increase rendering speeds, the image data (both DEM and imagery) is discretized into uniform regions of smaller tiles (Fig 4). The AGRI-AG texture management component selects appropriate resolution tiles using Level of Detail (LOD) methods. The tile selected is based on the distance between the viewer (camera) and the land tile as illustrated in Fig. 5. Methods for streaming and downloading the tiles are presented in section 2.4.

Insert Figure 4 & 5

2.3.2 3D Yield Map Visualisation

A 3D yield surface can be used to convey the heterogeneity in crop yield by both colour and/or height. The yield data is used to extrude the pixels based on the crop yield value. This generates a 3D surface where low and high heights correspond to low and high yields respectively. Fig. 6 shows the 2D and 3D yield maps for comparison. The shading model uses a lookup table of pseudo-normals to increase the rendering speeds during visualisation. This was implemented primarily as an optimization method for running the app on lower-end mobile tablets, as GPU does not need to compute the vertex normal directions every frame.

Insert Figure 6

2.3.3 3D Spatially Averaged (Aggregated) Data Visualisation

One way to visualise large amounts of quantitative spatial data is using spatial averaging methods (Spence, 2001). The number of yield data points to average are specified and the appropriate block/area size is then calculated. The data is assumed to be homogeneously distributed, which is a fair assumption for this type of data. At the centre of each block a 3D cuboid is generated, the height of which is scaled by the calculated averaged yield value. Other geometrical primitives can be used such as cones, spheres or cylinders. The aggregated data is read as raw data from a *Coma Separated Value* (CSV) data file, which can be downloaded from the server, and includes the yield, latitude and longitude values. The fewer points per bock will result in more 3D object primitives to be displayed (Fig 7). The aggregated 3D visualisation method also uses “pseudo-normal” calculations for surface shading. Therefore, all 3D objects have the same facing vertex normals thus they are all lit and shaded in one direction.

Insert Figure 7

2.4 Texture Streaming Methods

Methods implemented in AGRI-AG for streaming high resolution data from the farm server include HTTP and WebSockets (Andersson and Göransson, 2012). HTTP is a default standard for data transfer between web connected applications on mobile devices, and WebSockets are currently becoming more widely used and are an already adopted standard (Grigorik 2013). HTTP based streaming makes use of "long polling" HTTP method where a connection to the server is established and the client requests data. After a set time-out period, the connection is closed and the client has to connect to the server again. Alternatively, WebSockets allow for a constant connection to be maintained between the client and server. WebSockets make use of bi-directional communication between the client and the server, and the connection is kept constantly open. Data transmission is considered to be low-bandwidth as the data packets are transmitted via the WebSockets protocol run on top of a single TCP connection (Grigorik, 2013). Fig. 8 illustrate how the HTTP long-polling and WebSockets communication works between the client and server

Insert Figure 8

2.5 AGRI-AP Performance Evaluation

2.5.1 Benchmarking of app and data visualisation techniques

As the aim was to ensure interactivity of the app two key performance indicators were measured for the different visualisation methods: Frames per Second (FPS) and the Milliseconds per Frame (MPFS). The RAM and CPU usage were also monitored. The mobile tablets used for testing were the Asus Google Nexus 7 and HTC Google Nexus 9 tablets. These tablet models were chosen because they provide a good range for comparison across the hardware capability spectrum. The Asus Google Nexus 7 tablet is an older generation Android mobile tablet with support for version 4.3 of the Android operating system (called "Jelly Bean"). It features a 1.51 GHz quad-core Krait 300 CPU, 2 GB DDR3L RAM and a Qualcomm 400 MHz quad-core Adreno 320 GPU. The HTC Google Nexus 9 tablet is a more powerful Android tablet featuring support for Android 5.0.1 (called "Lollipop"). The Nexus 9 features a NVIDIA Tegra K1 CPU (2.3 GHz dual-core 64-bit "Denver"), 2 GB LPDDR3-1600 RAM and a NVIDIA Kepler GPU. The most significant difference between the two Nexus 7 and 9 tablets is the support for 3G/4G mobile networking supported only by the Nexus 9. All of the profiling was done using the ADT debug tools within the Eclipse integrated development environment (IDE).

2.5.2 Data Streaming

Two use cases were selected for evaluating the streaming methods: high connectivity (via Wi-Fi) and low connectivity (via 3G). Testing the steaming in these two environments reflected the conditions in which the app would be used. The chosen high-connectivity environment was the Abertay University campus and the streaming methods were tested using a standard Wi-Fi network connection. The chosen low-connectivity environment was Tentsmuir Forest in Fife, Scotland (see Fig. 9).

Insert Figure 9

The HTC Google Nexus 9 was used as the main tablet for the low and high-connectivity environment testing. The Google Nexus 9 tablet was used as it features support for 3G/4G mobile communication, which is essential for testing in the field. The streaming testing protocol included downloading a single large 2048x2048 compressed JPEG image tile for a given DEM tile region and recording the time to download.

2.5.3 User evaluation

A focus group was set up to determine the user perceptions of the different visualisation techniques. The focus group was recruited to reflect the potential user base and included digital and non-digital natives. The focus group involved downloading the app on the user's own devices and trialling the functionality and visualisation methods. The qualitative testing focussed on usability, visual preference and overall impact – which participants worked through at their own pace. There were eight participants in the user testing group, which included farmers, agronomists, PA technologists and academics. The questionnaire is presented in App 1.

3. Results

3.1 Data Visualisation techniques

Fig. 10 shows the results of the visualisation techniques implemented to display the yield data for a given field. The 2D colour coded map Fig. 10 a) is the most familiar style to farmers and agronomists.

Insert Figure 10

Fig. 11 shows an "exploded view" of time series yield data for the same field. This emphasises the customisability of visualisation of AGRI data afforded by the programmable pipeline on mobile devices. Alternative methods of animating time series data is shown in Fig. 12.

Insert Figure 11

Insert Figure 12

3.2 Performance Results

Fig. 13 - 16 show the average FPS, MFPS, RAM usage and CPU usage results for each of the visualisation methods tested on the Nexus 7 and 9 tablet devices. Higher FPS values indicate better rendering performance, while smaller MFPS value indicate higher rendering efficiency (less time spent) rendering each frame. Lower CPU and RAM usage values are preferred. Each of the performance tests were replicated 15 times to obtain a distribution. The data is

not norammaly distributed therefore error bars are not presented on the charts. For the aggregated data visualisation methods, the point sample size of 10, 30 and 50 was chosen for the benchmarking which results in 1546, 537, 338 3D primitives to render. RAM usage is far lower on the Nexus 9 than on the Nexus 7 due to the use of the new runtime ART VM, which has more optimizations than the previous VM version Delvik which is used by the Nexus 7. Interactive performance on the Nexus 9 is slightly worse than on the Nexus 7. This is because the application was developed originally for version 4.3 of the Android operating system running on the Nexus 7 tablet. Nexus 9 uses version 5 of the Android operating system (called “Lollipop”) and also uses a completely re-designed version of the runtime virtual machine (VM) called ART (Toombs, 2013). The code was not ported nor optimized specifically to make use of any of the new features of version 5 of the Android operating system.

Insert Figure 13

Insert Figure 14

Insert Figure 15

Insert Figure 16

3.3 Texture Streaming Results

The time taken to download the 2048x2048 compressed JPEG texture image using HTTP and Web Sockets in a high and low connectivity environment is presented in Fig. 17. The connectivity results show that in a high connectivity environment, the use of WebSockets for streaming on the Nexus 9 tablet is significantly faster in comparison to HTTP-based streaming (see Fig. 17). Testing in a low-connectivity environment was performed using only the Nexus 9 tablet as it features support for 3G communication. The results obtained from the low-connectivity environment show that the usage of WebSockets-based streaming is faster than HTTP-based streaming. The performance variances found in the WebSockets-based streaming method using the 3G network connection protocol are due to non-standardized support for WebSockets over the 3G communication network. This has been researched and reported by (Estep, 2013), and his research concludes that WebSockets performance can vary significantly depending on the network communication protocol that is being used.

Insert Figure 17

3.4 Qualitative User Testing Results

The app was tested by exploring and monitoring crop yields of a single field over time and with different presentation modes. A summary of the testing together with some statements from users is presented The user interface was described as having a clean layout and graphical style but there was however some issues with the navigation being non intuitive. The users requested both gesture based navigation and a navigation wheel such as in Google maps. With regard to visual preference the users found that the use of aerial photography overlaid on top of a 3D digital elevation model was beneficial for contextualising the main features (e.g. farm fields, buildings, lochs). It was also noted that the texture resolution should be higher and more crisp when the user zooms in. Most users rated the two interpolated crop yield data time series renders with the highest preference. A suggestion was made to include an "exploded view" of the yield data for the different years, as well as the ability to playback and through time series using a video-like playback interface. These features have been added for the final release version of the app as shown in Fig. 11 and 12. The overall impact section revealed that users were generally satisfied with the app, but improvements could be made by incorporating other data such as chemistry (PH, nutrient), soil values / soil texture, rainfall per week. One test participant wrote on the feedback form: "*Both methods* (top down and 3D view) *of the land area are useful. What I like most is that the 3D terrain model could show field terrain better than 2D* (map)." Some users found the 3D spatially averaged visualisation method to be particularly engaging, especially when compared to the 2D yield maps. Another participant stated that what they liked most about the app was "*rapid visualisation of yield data*", but that they disliked the "*3D view of aggregated data*". The ability to animate through time series data was also positively received. The users found it useful to switch between the visualisation methods seamlessly and in real-time. One tester stated in the feedback that "*The tilted top-down view is easier to see and to control but that a top-down view is also useful in certain scenarios. (App) doesn't seem to have noticeable performance hits (when viewing terrain) and greatly aids the user in determining where they are looking*". Reservations were made about the lack of gesture based scene navigation, the method for zooming in and out of the scene (as this was tied to button controls rather than gesture based controls). One tester commented that "*Buttons have confusing terminology (names) and that vertical axis rotation is opposite to what I expect.*", and another mentioned that "*A reset button for navigation should be added along with gesture based control*" and that "*pinch (zoom) function would be nice*". The navigation control issues were addressed and changed to complete gesture based control after the feedback was provided.

4. Discussion

The research findings have shown that high-resolution aerial photography and crop yield data can be streamed from a remote server and displayed in an interactive context on mobile devices. It is shown in both low and high connectivity environments that WebSockets are significantly faster than using HTTP-based streaming. WebSockets make use of bi-directional communication between the client and the server. The connection between the server and the client is kept alive throughout the communication period. Therefore data can be transmitted between the client and the server simultaneously without opening and closing the connection. This makes the WebSockets communication protocol comparable to low-latency network data transfer and has increased the protocols popularity for use in applications that require low-latency real-time communication (Grigorik, 2013).

Further the application performance results show that the implemented visualisation methods can be rendered in real-time. The issues highlighted by Chen et al 2015 with respect to data analysis and presentation being a bottleneck in PA can to some degree be overcome with the presented framework. The varying preferences with respect to visualisation techniques further support that a suitable way forward is providing the users with a selection of methods to choose from. It is suspected that those that are used to 3D visualisation and considered digital natives may find the 3D methods more intuitive whilst others do not. The flexible customisation of data presentation, achieved by programmable pipelines, is useful for a large user base where new ‘effects’ can be tried out.

Improved 3D experience on mobile phones is set to revolutionize the multimedia market and a key challenge is identifying useful 3D visualization methods and navigation tools that support the exploration of data driven 3D interactive visualisation frameworks.

5. Conclusion

The developed AGRI-AG application demonstrates that mobile devices are capable of streaming and displaying 3D maps of farm AGRI data, in novel ways on commodity mobile devices, within an interactive 3D context. This may benefit stakeholders in terms of enhanced engagement and delivery of context-aware and relevant data. Different data visualisation techniques have been described, implemented and assessed for presenting farm data and the wider geographical context. The power consumption and the effect AGRI-AG has on the mobile device battery life was not determined. Extensive in-field testing of the application to specific agricultural tasks is also part of future work. The AGRI-AG application can be improved by having better integration with web database services for accessing aerial imagery and geospatial data in real-time as well as for uploading data to a farm server. The core platform can be applied to many other spatial data-rich sectors including environmental monitoring and homeland security.

References

Akenine-Möller, T., Haines, E., & Hoffman, N. 2008. Real-time rendering. CRC Press.

Andersson S. and Göransson, J. 2012. Virtual Texturing with WebGL. <http://publications.lib.chalmers.se/records/fulltext/155126.pdf>

Brothaler, K. 2013. OpenGL ES 2 for Android. The Pragmatic Programmers.

Burigat, S., & Chittaro, L. 2007. Geographical Data Visualization on Mobile Devices for User's Navigation and Decision Support Activites. In Spatial Data on the Web (pp. 261-284). Springer Berlin Heidelberg.

Butler, D. 2006. Virtual globes: The web-wide world. Nature, 439(7078), 776-778.

Calhoun, D. 2011. When to Base64 Encode Images (and When Not To). <http://davidbcalhoun.com/2011/when-to-base64-encode-images-and-when-not-to/>

Chehimi, F., Coulton, P., & Edwards, R. 2005. Evolution of 3D games on mobile phones. In Mobile Business, 2005. ICMB 2005. International Conference on (pp. 173-179). IEEE.

Chen, N., Zhang, X., & Wang, C. (2015). Integrated open geospatial web service enabled cyber-physical information infrastructure for precision agriculture monitoring. *Computers and Electronics in Agriculture, 111*, 78-91.

Chiles, J. P., & Delfiner, P. 2009. Geostatistics: modeling spatial uncertainty (Vol. 497). John Wiley & Sons.

Chittaro, L. 2006. Visualizing information on mobile devices. Computer, 39(3), 40-45.

Dunlop, R. 2005. FPS Versus Frame Time. MVPS.

Estep, E. 2013. Mobile html5: efficiency and performance of websockets and server-sent events. School of Science Double Degree Programme NordSecMob, Aalto University.

Falcao, A. O., dos Santos, M. P., & Borges, J. G. 2006. A real-time visualization tool for forest ecosystem management decision support. Computers and electronics in agriculture, 53(1), 3-12.

Falconer RE, Isaacs, JP, Blackwood, DJ, & Gilmour, DJ. .2015. The role of interactive visualisation in developing inclusive technology for sustainability of built environment. Carlos Nunes Silva (Ed.). Emerging Issues, Challenges, and Opportunities in Urban E-Planning. Hershey, PA: IGI Global. doi:10.4018/978-1-4666-8150-7

Falconer, R. E., & Houston, A. N. 2015. Visual Simulation of Soil-Microbial System Using GPGPU Technology. Computation, 3(1), 58-71.

Fountas, S., Carli, G., Sørensen, C. G., Tsiropoulos, Z., Cavalaris, C., Vatsanidou, A., ... & Tisserye, B. (2015). Farm management information systems: Current situation and future perspectives. Computers and Electronics in Agriculture, 115, 40-50.

Ge, R., Feng, X., & Cameron, K. W. 2005. Improvement of power-performance efficiency for high-end computing. In Parallel and Distributed Processing Symposium, 2005. Proceedings. 19th IEEE International (pp. 8-pp). IEEE.

Grigorik, I. 2013. High Performance Browser Networking: What every web developer should know about networking and web performance. O'Reilly Media, Inc.

GS+ : Geostatistics for the Environmental Sciences. (2016). Gammadesign Software. Available online: https://www.gammadesign.com/ . Last Accessed on 22/09/2016.

HM Government. (2013). A UK Strategy for Agricultural Technologies. Avialable online: https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/227259/9643-BIS-UK_Agri_Tech_Strategy_Accessible.pdf - Last Accessed 22/02/2017)

Hopkins, M. 2013. 20 Best Mobile Apps for Agriculture. <http://www.croplife.com/editorial/matt-hopkins/20-best-mobile-apps-for-agriculture/>

Isaacs, J. 2011. Sustainability Assessment and Visualisation in Urban Environments. PhD Thesis. University of Abertay Dundee.

Isaacs, J. P., Blackwood, D. J., Gilmour, D., & Falconer, R. E. 2013. Real-Time Visual Simulation of Urban Sustainability. International Journal of E-Planning Research (IJEPR), 2(1), 20-42.

Karetsos, S., Costopoulou, C., & Sideridis, A. 2014. Developing a smartphone app for m-government in agriculture. Agrárinformatika/Journal Of Agricultural Informatics,5(1), 1-8.

Koskela, T., & Vatjus-Anttila, J. 2015. Optimization Techniques for 3D Graphics Deployment on Mobile Devices. 3D Research, 6(1), 1-27.

Lange, E. 2001. The limits of realism: perceptions of virtual landscapes. Landscape and urban planning, 54(1), 163-182.

López-Riquelme, J. A., Pavón-Pulido, N., Navarro-Hellín, H., Soto-Valles, F., & Torres-Sánchez, R. (2016). A software architecture based on FIWARE cloud for Precision Agriculture. Agricultural Water Management.

Lovett, A., Appleton, K., Warren-Kretzschmar, B., & Von Haaren, C. 2015. Using 3D visualization methods in landscape planning: An evaluation of options and practical issues. Landscape and Urban Planning, 142, 85-94.

Lunden, I. 2014. Gartner: 195M Tablets Sold In 2013, Android Grabs Top Spot From iPad With 62% Share. <http://techcrunch.com/2014/03/03/gartner-195m-tablets-sold-in-2013-android-grabs-top-spot-from-ipad-with-62-share/>

Martinez, K., & Cupitt, J. 2005. VIPS-a highly tuned image processing software architecture. In Image Processing, 2005. ICIP 2005. IEEE International Conference on (Vol. 2, pp. II-574). IEEE.

McLachlan, P. 2013. On Mobile, Data URIs are 6x Slower than Source Linking (New Research). < http://dev.mobify.com/blog/data-uris-are-slow-on-mobile/>

Mulla, D. J. 2013. Twenty five years of remote sensing in precision agriculture: Key advances and remaining knowledge gaps. Biosystems Engineering, 114(4), 358-371.

Munshi, A., Ginsburg, D., & Shreiner, D. 2008. OpenGL ES 2.0 programming guide. Pearson Education.

NetMarketShare. 2016. Mobile/Tablet Operating System Market Share. <https://www.netmarketshare.com/operating-system-market-share.aspx?qprid=8&qpcustomd=1>

Phillips, B., & Hardy, B. (2013). Android Programming: The Big Nerd Ranch Guide. Pearson Education.

Pombinho, P., Ana, M., Afonso, P., Carmo, M. B., & Grande, C. 2007. Geo-referenced Information Visualization on Mobile Devices.

Popov, S. 2009. When you should use base64 for images. <http://www.stoimen.com/blog/2009/04/23/when-you-should-use-base64-for-images/>

Santana, F. S., Murakami, E., Saraiva, A. M., & Correa, P. L. 2007. A comparative study between precision agriculture and biodiversity modelling information systems. In World Conference on Computers in Agriculture and Natural Resources (EFITA) (pp. 1-6).

Shebanow, M. 2013. An evolution of mobile graphics. Keynote talk at High Performance Graphics.

Siart, C., Bubenzer, O., & Eitel, B. 2009. Combining digital elevation data (SRTM/ASTER), high resolution satellite imagery (Quickbird) and GIS for geomorphological mapping: A multi-component case study on Mediterranean karst in Central Crete. Geomorphology, 112(1), 106-121.

Siebert, S., & Teizer, J. 2014. Mobile 3D mapping for surveying earthwork projects using an Unmanned Aerial Vehicle (UAV) system. Automation in Construction, 41, 1-14.

Spence, R. 2001. Information visualization (Vol. 1). New York: Addison-Wesley.

Thiagarajan, N., Aggarwal, G., Nicoara, A., Boneh, D., & Singh, J. P. 2012. Who killed my battery?: analyzing mobile browser energy consumption. In Proceedings of the 21st international conference on World Wide Web (pp. 41-50). ACM.

Tiffin, R. (2017, February). Agrimetrics: The World's First Big Data Center for the Agri-Food Industry. In 2017 AAAS Annual Meeting (February 16-20, 2017). aaas.

Toombs, C. 2013. Meet ART, Part 2: Benchmarks - Performance Won't Blow You Away Today, But It Will Get Better. Android Police. http://www.androidpolice.com/2013/11/12/meet-art-part-2-benchmarks-performance-wont-blow-away-today-will-get-better/

Vlissides, J., Helm, R., Johnson, R., & Gamma, E. (1995). Design patterns: Elements of reusable object-oriented software. Reading: Addison-Wesley, 49(120), 11.

Whitacre, B. E., Mark, T. B., & Griffin, T. W. 2014. How Connected are Our Farms?. Choices, 29(3).

Ying, F., Mooney, P., Corcoran, P., and Winstanley, A. C. 2012. Dynamic Visualization of Geospatial Data on Small Screen Mobile Devices (pp. 77-90). Springer Berlin Heidelberg.

Zhang, N., Wang, M., & Wang, N. 2002. Precision agriculture—a worldwide overview. Computers and electronics in agriculture, 36(2), 113-132.

.

Appendix A –User Testing Questionnaire

Insert Appendix B Image 1

Insert Appendix B Image 2

Insert Appendix B Image 3

Insert Appendix B Image 4

Insert Appendix B Image 5

Insert Appendix B Image 6

Acknowledgements

The authors would like to thank SAGES, Innovate UK and Soil Essentials Ltd for their financial support and collaboration in this project.

Figure Captions

Figure 1: Example integration of the AGRI-AG app, and related app inputs, into an existing FMIS. Adapted from Fountas et al 2015.

Figure 2: Flowchart diagram illustrating the key components of the AGRI-AG framework.

Figure 3: The yield map generation process using offline pre-processing. The offline generation of yield maps is done using GS+ software, where the generated yield map along with the colour table are exported as image files to be used in the main AGRI-AG interactive visualisation scenario. The yield map image data can be used for both 2D and 3D projections for visualisation purposes of stakeholder engagement.

Figure 4: Example of the LOD tile selection for AP imagery used in AGRI-AG.

Figure 5: Correspondence between a) the location of the selected farmland area, b) tiled aerial photography image data and c) the tiled digital elevation model.

Figure 6: Visual differences between the 3D (left) and 2D (right) yield map visualisation methods.

Figure 7: The spatially averaged algorithm using 1 point per block shown within the 3D context and showing Lat Long coords in top left. The GUI layout is also shown.

Figure 8: Illustration showing communication between client device and server using *a)* HTTP Long-Polling based and *b)* WebSockets-based connectivity methods.

Figure 9: Pictures from the low-connectivity testing site in Tenstmuir Forest, Fife, Scotland.

Figure 10: Examples of the three data visualisation techniques, *a)* 2D map *b)* 3D map and *c)* aggregated 3D visualisation

Figure 11: Examples exploded display of yield time series data.

Figure 12: Frames of a time animation of the yield data

Figure 13: Average FPS performance result.

Figure 14: Average MFSP performance result.

Figure 15: Average RAM usage performance result.

Figure 16: Average CPU usage performance result.

Figure 17: High and low-connectivity environment testing results on the Nexus 9 tablet. The milliseconds correspond to the elapsed image texture download time.