A Sensor Based 3D Annotation Authoring Tool for Outdoor Field Applications

Mustafa Tolga Eren
Computer Science and Engineering Program,
Sabanci University
Istanbul, Turkey
mustafae@sabanciuniv.edu

Selim Balcisoy
Computer Science and Engineering Program,
Sabanci University
Istanbul, Turkey
balcisoy@sabanciuniv.edu

***Abstract*— Many professional tasks such as geographic or archaeological surveying require editing and processing spatial data. User generated models act as an underlying level for visualizing annotations. For accurate and fast placement of annotations in the field we introduce a mobile modelling workflow. Our contribution includes i) a novel annotation technique based on geometric regions and ii) a modelling workflow optimized for mobile devices.**

Image-based Modelling, Annotation, Mobile Graphics

I. INTRODUCTION

Outdoor field works such as geographic or archaeological surveying require editing and processing of semantic information on spatial data. Currently these studies are performed manually using pen and printed maps or a laptop with a GPS receiver and digital two dimensional (2D) maps. In many cases such as rescue excavations for urban archeology or site surveys after a flooding; there is limited time before construction work starts and traditional techniques do not suffice.

In practice annotations are used to mark different layers and regions in civil engineering or stratigraphy studies. The processing time consists of manual work in the field and digitization of the annotations at the office afterwards. A laptop allows users to process digital data in the field but hinders walking around freely and requires constant switching between the laptop screen and the real world. This mental mapping process may lead to high error rates. In this paper we demonstrate improving the limitations of this workflow using hand-held mobile computers.

Hand-held mobile computers already have started replacing notebooks and desktops for many computing tasks in the field. These devices have several shortcomings, such as: limited battery life, small display area and limited user interaction. Solutions which have been optimized for desktop environment need to be carefully re-designed and extended in line with the requirements of the mobile work environment. The main goal of this paper is to let the professionals perform the annotation task in the field successfully using a mobile device in minimum time.

We propose a workflow featuring a simple modelling and annotation authoring process. There are two major issues need to be taken into consideration; i) how to create three dimensional (3D) annotations and ii) how to visualize these annotations in a mobile context. An annotation can be defined as adding extra virtual information over a real object [30]. We extend this definition and employ a variety of annotation types ranging from a single point to four dimensional (4D) annotations, an annotation of a volume over time. In the context of this paper, the main goal of annotations is to identify the primary building blocks or layers of an object. Labels and text may not be enough for complete annotation authoring. Archaeologists and civil engineers are interested in layer based studies such as stratigraphy. In order to annotate a layer of a 3D object correctly, a 2D label is not the best choice. A layer represents a volume of the object, so we propose a volume based annotation authoring process.

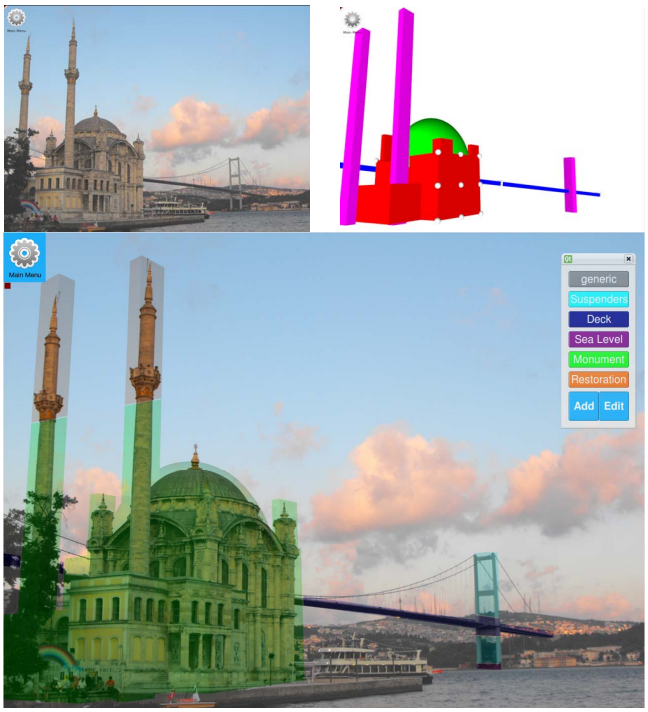


Figure 1. Example An urban scene is (a) photographed. Using these images, two objects are (b) modelled and (c) annotated using our workflow. Annotations are color coded; a legend is shown in the canvas for identification.

978-0-7695-4467-0/11 $26.00 © 2011 IEEE
DOI 10.1109/CW.2011.39

To visualize annotations, we utilize user generated 3D models as an underlying structure. These models are fast generated, and roughly represent the object. We do not require high precision models as we only need these models to lay annotations over them.

When dealing with static images, where the user only observes the scene from a single point, annotation authoring and management can be achieved using 2D constructs such as labels and floating text [16]. On the other hand, in mobile context the user can freely move in the scene, thus effectively changes the eye position. When the user moves, 2D constructs may start overlap or even become distorted and very hard to read. In order to handle annotation visualization for mobile users, an underlying 3D structure is preferred, especially to handle occlusions by utilizing depth information [15]. Kopf et al. used high quality models and accompanying textures to visualize and annotate large scenes, such as Manhattan Island. Although the results are visually impressive, editing and processing of dense models on a mobile device may not be feasible. In addition, our use cases require fast-generated and simple models that allow editing for annotation authoring. To overcome these issues, we propose a simple and semi-automated image-based modelling process, where the user combines several building blocks in order to create a model. Annotations are then presented over these user generated models.

Our contribution includes i) a novel annotation technique based on 3D geometric regions and ii) a fast modelling workflow based on building blocks. To best of our knowledge, this is the first method that utilizes volumetric annotations. We also introduce a simple and intuitive interface for modelling and annotation editing processes. In addition, we conducted a user study and observed that the proposed solution is suitable for mobile field work.

II. Related Works

Modelling: Object modelling is a well-researched topic in both computer graphics and vision. Geometric models can be created from scratch or sampled from real objects using a number of techniques. Many commercial 3D modelling packages support image based modelling tools, such as Blender [5]. These packages often support using top, side and front photograph views as superimposed over the model. There are also fully automated solutions based on computer vision techniques for creating models out of sets of images [21]. However these are prone to artifacts caused by vision algorithms when fed with noisy or under exposed images. In order to deal with these artifacts researchers adopted semi-automated processes such as PFTRACK and Vodoo [7, 18, 26].

These approaches allow some user interaction; i.e. letting users to manually mark corresponding features. VideoTrace by van den Hengel et al. [29] is an improvement over semi-automated processes as it supports user interacted geometry creation, however it requires users to work within the VideoTrace environment. Like VideoTrace, Sinha et al. [24]'s system makes use of the underlying sparse reconstruction, moreover they utilize vanishing directions. Recently Thormählen and Seidel [27] presented an ortho-imaged based solution for creating high quality models without forcing modellers to leave their desired modelling environment. Other vision-based methods use large geo-tagged photo sets to generate textured 3D models of buildings [9, 25, 32].

Annotations: Annotating real objects is heavily investigated under Augmented Reality (AR). Feiner et al. [8] and Rekimoto and Nagao [22] were early works used AR to annotate the real world with overlaid textual labels. Although a 3D model is generally used to place annotations, Snavely et al. [25] used a system to transfer annotations from one image to another. Recently Wither et al investigated annotations in outdoor augmented reality domain [31, 30]. Another outdoor AR work; by Schall et al. [23], introduced an annotation authoring tool which creates 2D information labels in 3D coordinates. Visualization of annotations is also a popular research topic. Annotations can be associated with a 2D point [1] or a 3D position [12] depending on the application. Generally if the virtual camera is mobile, the 3D approach is preferred.

Mobile Studies: Mobile Studies: Our modelling approach is inspired by image-based methods. Similar approaches have been utilized by Piekarski [19] to create object models in the field using a backpack based system known as Tinmith-Endavour. MARS is another backpack based system which also includes a hand-held device to annotate and view merged environments [13]. To author physical models, Baillot et al. [2] used mobile computers by generating 3D models from floor plans via user interaction. Backpack-based approaches offer computing power as well as centimeter accurate GPS sensors.

Although a backpack-based computer was required for these tasks in the past, currently hand-held computers are capable of performing even more complicated tasks[28]. A recent work by Schall et al. [23], focuses on displaying pre-defined 3D models to aid civil engineers using hand-held mobile devices. For on-site archaeological studies Benko et al. [4] provided collaborative mixed reality visualization following data recording and archiving principles defined by Harris [10].

III. Modelling

Our modelling process utilizes a "construction toy" analogy. The output of our modelling process is a combination of interlocked primitives. In order to create a complex model, user attaches 3D geometric primitives to each other, one at a time. These 3D geometric primitives are referred as "building blocks" in the rest of the paper. For simplicity the variety of building blocks are kept at minimum, i.e. cube, column, dome and cone. However for each building block the user is able to define an independent transformation. By utilizing these individual transformations it is possible to create many required primitives to model a building. Semi-automated approaches have long been examined for image-based modelling processes.

In many of these approaches, the user is asked to match exact features in several images [11]. More recently, VideoTrace[29], allowed users to define polygons on video frames and these polygons are auto transformed with respect

to camera positioning. Our approach lies in between; rather than letting the user match exact positions in several images, we ask the user to adjust an initial building block, incrementally fixing the orientation over several images. Additional building blocks inherit the orientation of this reference block. The final orientation is saved in real world coordinate system. Using GPS and digital compass data associated with every reference image, we triangulate and find the estimated position for the real world objects. This is done via casting rays from the image location, along the heading direction. In an optimal environment, the target object's location would be where the rays intersect. However due to noisy sensor data, instead of a fixed location we may end up with an area. In this case we assume the target's location is inside this area, and use the geometrical center of it as the location.

The modelling process starts with inserting an initial building block to our scene. This block is translated, rotated and scaled by the user, to match a primitive of the object that is being modelled. Then user is able to drag and drop next desired building block to the scene. The new block is attached to the model when the user drops this building block on to any previous block. In this case the newly added block is automatically transformed and inserted to the scene hierarchy as a child of that previous block. The user may adjust the transformation via a simple graphical user interface (GUI). This process is repeated until the object is completely modelled.

The Building blocks can interlock each other at 26 different locations. These locations lay on the bounding box of each block. They consist of 8 corner points, 12 points in the middle of each corner pair and 6 face middles.

The interlocking process takes source and target blocks' scale and an interlocking vector as input. For example if the user inserts a new block to right side of a previous block, then the interlocking vector should have a positive value along the X axis, in particular this vector is v(1;0;0). To adjust the scale of the new block, the axes with 0 value is considered. Corresponding scale values on these axes are used to find the maximum ratio in between. The inverse of this ratio is used to scale down the new block. After auto-scaling, the new block is translated to the edge of the previous block to make the blocks look like interlocked at each other. A newly added block carries the rotation of its parent.

To create holes and extrusions on the model, the user is provided with fine tuning tools such as slicing and extrusion. Slicing is achieved via adding user defined vertices on to a plane on the model. The newly added vertices along with the initial vertices of the plane, then fed to a constrained Delaunay Triangulation. This process is demonstrated in Figure 2.

The output of this triangulation is the same plane with more polygons in it. The user is able to delete any of these polygons to create holes, or extrude them to create additional extrusions. During the modelling process photographs of the modelled object is shown as background images.

The virtual camera is translated to relevant position for each corresponding image. By utilizing a pre-computed camera calibration, the model is ensured to superimpose the object correctly for each image.

When modelling is completed it is possible to export geometric data into a Collada [3] supported format.

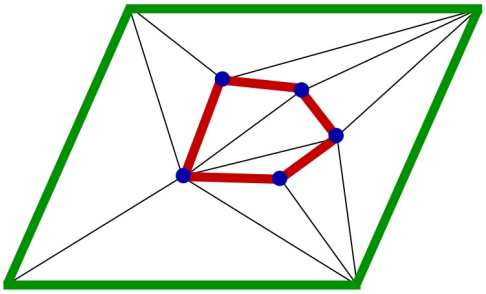


Figure 2. Example The green polygon is the initial polygon. The red polygon is defined by user clicks. Green and red edges are supplied to Delaunay triangulation. The output is the combination of green, red and black edges.

IV. ANNOTATIONS

Wither et al. explains that every annotation should have two parts; a spatially dependent component that links to the object and a spatially independent component that contains the information that is to be annotated over the object [30].

The annotation system presented in this paper is based on the definition of Wither et.al. We extend this definition by adding specific items for spatial and semantic components. These components vary as detailed in Tables 1 and 2. Spatial component of an annotation is defined as one of the following; vertex-based, planar or volumetric. Semantic Component can have all of the values described in Table 2. Using a combination of these components it is possible to create any annotation ranging from a label to a 4D annotation, an annotation of a volume over time.

In order to create an annotation the user first defines a spatial component and assigns a semantic component to it. The semantic component can be previously defined or can be created from scratch on-the-fly.

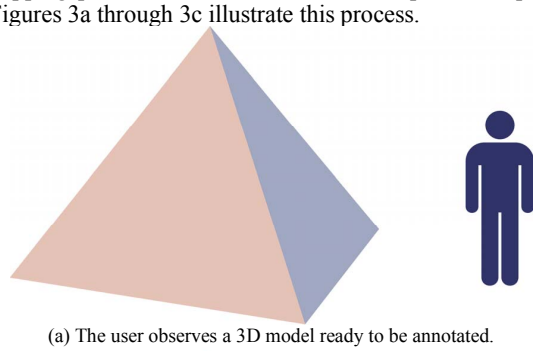
A. Spatial Component

There are three different geometric options for defining the spatial component using the previously generated model. In case of vertex-based spatial component, the user simply defines a point in the scene by clicking to the desired location. A label is created in this location, representing semantic component of this annotation.

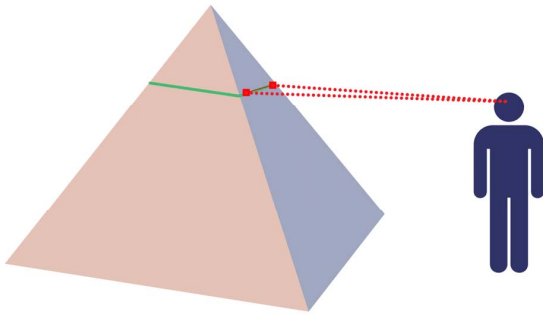
For a planar spatial component, the user is able to select a face of any building block. It is also possible to adjust this selection by adding arbitrary points on the face to create a more detailed polygon on the model. This is achieved by inserting user defined vertices on the face and computing a constrained Delaunay triangulation.

In order to create a volumetric spatial component, the user needs to define a volume on the model. This process is simplified by utilizing clipping planes. The user creates desired number of clipping planes to create a sub-section of the 3D model. The volume which resides in between the

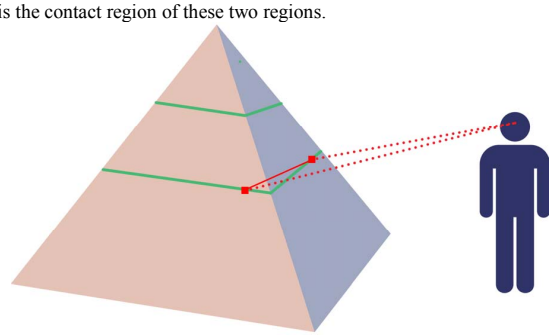
clipping planes becomes the volumetric spatial component. Figures 3a through 3c illustrate this process.



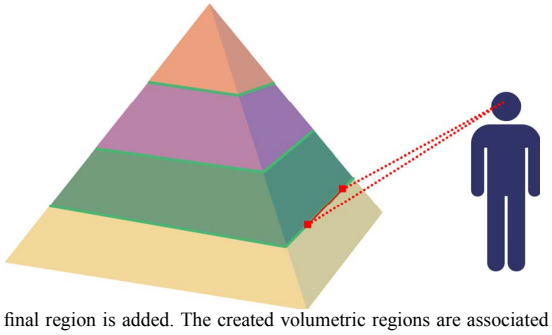
(a) The user observes a 3D model ready to be annotated.



(b) Red squares denote user clicked 3D positions. Using these two points and the position of the virtual camera, a clipping plane is calculated. With this clipping plane the 3D model is divided into two 3D volumetric regions. Green line is the contact region of these two regions.



(c) A new volumetric region is generated using the same approach in Figure 3b. The user clicked points, do not have to be on the same face. As long as they are located on the model geometry, a new clipping plane is calculated.



(d) A final region is added. The created volumetric regions are associated with semantic components to create annotations. The annotations are presented in different colors and superimposed over the model.

Figure 3. Annotation creation process is illustrated.

TABLE I. SPATIAL COMPONENT

Spatial Component	***Description***
Vertex-based	A point in the scene
Planar	A plane on the model
Volumetric	A volume of the model

B. *Semantic Component*

A semantic component must have an ID and a color, other fields are optional. When assigned to a vertex-based spatial Component, an annotation is created as a label. The ID of the semantic component is displayed on this label with the appropriate color. When assigned to a planar or volumetric spatial component, the geometric region defined by the spatial component is colored accordingly to create an annotation as seen in Figure 3d. A legend is also displayed to identify colored components on the canvas separately.

We utilize time as a variable to visualize annotations, in a chronologically ordered scene. As shown in Figure 1, many urban settings contain visible objects from different eras; the user is able to observe annotations of these objects in chronological order by moving a time slider. As time progresses, relevant annotations simply fade in to the scene to superimpose the real world images. The annotation is active and visible only for the interval defined in the associated semantic component. A descriptive text is shown when the user clicks a specific annotation.

TABLE II. SEMANTIC COMPONENT

Spatial Component	***Description***
ID	A name
Color	RGB color values
Text	A decription
Time	A time interval

V. WORKFLOW AND DESIGN CHOICES

This chapter elaborates on the design choices we have taken. A flow chart demonstrating our approach can be seen in Figure 4. Figures 5 to 8 demonstrate the workflow with specific examples. The first step of our process is capturing and placing images in our scene coordinate system. This requires GPS and heading data. The minimum required number of images is one, however capturing 2-5 images from different viewing angles produces better results. These images will be used as reference images in the application.

The next step is modelling. Reference images serve as background and virtual camera is translated in order to represent the position of the real camera. The very first building block for each new object establishes a mapping from scene coordinates to object coordinates. We call this the reference block of the object. This reference block is the root of the object building block hierarchy. We save the orientation information and use it to place our model in world coordinates by a simple triangulation process. The user can also translate the virtual camera to a pre-defined

position such as top view. This is similar to the approaches used by [20]. The user is now able to add new blocks to the scene using point and shoot analogy. The new block is attached to the user selected block along the interlocking vector. The interlocking vector is selected by clicking directly on the building block's related area. Alternatively a pre-defined vector can be selected from the GUI.

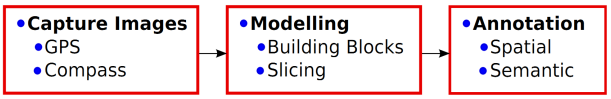


Figure 4. Our workflow is summarized in three steps. A modelled object can be annotated more than once.

Figure 5. A building is photographed from four different angles, two of these are shown here.

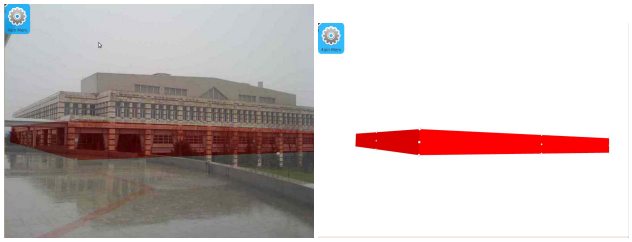


Figure 6. Modelling process starts with creating and adjusting a reference block. This block has the same orientation with the building.

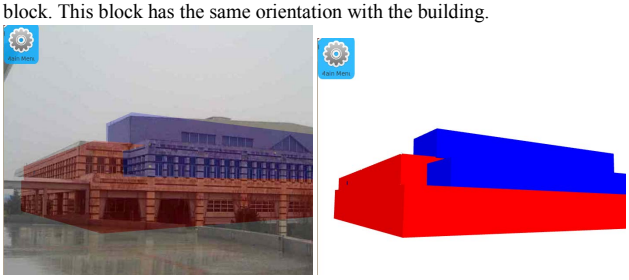


Figure 7. Completed model is shown; in this example 6 blocks are used to model the entire building.

Figure 8. After generating volumetric regions as spatial components, four different annotations are created. These are, from top to down; 2nd Floor, 1st Floor, Ground and Basement.

It is also possible to define a volume of the model as spatial component. In order to define a sub-section of the model as a volume, the user facilitates clipping planes. She clicks two different points on screen to form a clipping plane. This plane divides the model into two different volumetric regions. Any number of regions can be created by defining additional clipping planes as shown in Figure 3. The volumetric region that lies in between consecutive clipping planes becomes the spatial component. This process is especially useful for defining layers in stratigraphic studies.

After identifying a spatial component the user assigns a semantic component to complete the annotation process. It is possible to use a pre-defined semantic component or create a new one. An ID and a color is required for each semantic component. A dialog window is used for creating and editing semantic components. This dialog window contains a color picker in RGB color space and input fields for related text and sliders for time.

A modelled object can be annotated more than once; i.e. for several users or may be updated to reflect recent changes.

VI. Discussion and Conclusion

We conducted a user study to test the ease of use our framework. The users were asked to model and annotate a historical building, as seen in Figure 9. In these tests, the average task completion time for modelling task is 647 seconds. In this period of time, users interacted with the touch screen approximately 320 times to model the building given previously captured images. All users produced usable models which can be correctly annotated, with 8.3 building blocks on the average.

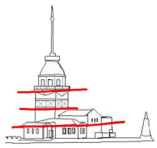
The average task completion time for annotation is 168 seconds. Users interacted with the touch screen approximately 36 times to create and label four different volumetric annotations. Only one, out of eight users, failed to generate these layers correctly. Users found our framework generally to be user friendly, a score of 4.1 out of 5, 5 being very user friendly, is received from qualitative questions.

For an average user it takes about 15 minutes from scratch to model and accurately annotate a building with approximately 10 blocks. 65 percent of all the touch screen interaction is navigation through images and menus. Reducing this ratio would result in even faster task completion times. As a future work, we are looking into multi-modal interfaces; voice and sensor-based interaction for simplifying navigation will be investigated.

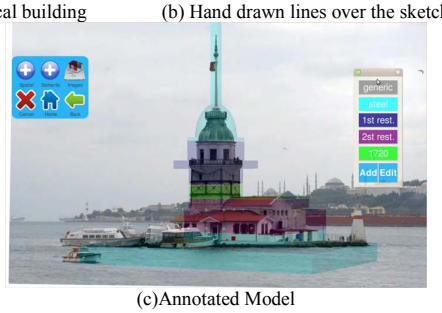
In order to compute physical effort for each task, we employ "interaction per minute" measurements. Average touch screen interaction per minute is about 29.6 for modelling and 12.8 for annotation tasks. Annotation authoring is performed only with 43 percent physical effort of modelling. The results are in line with our expectations that a geometric 3D annotation method can be used with mobile systems, if the interactions are properly designed.

Our volumetric annotation system is most applicable to layer based identification. This identification method is mainly used in stratigraphy and archaeological studies. It is possible to include different annotation schemes by simply registering extra clipping planes for regions.

(a) A historical building



(b) Hand drawn lines over the sketch



(c)Annotated Model

Figure 9. The real world image(a), is annotated using our workflow(c). The sketch(b) is provided to subjects as a guideline for annotation task of the user study. Subjects were expected to label four different layers, namely; steel support, first restoration, second restoration and new base.

Each modelling tool has its strengths, some create highly accurate visuals [27] and some others emphasize on fast modelling [14, 28]. Modelling is essential in our workflow in order to visualize annotations in a meaningful way. Our modelling flow has a simple and intuitive interface for modelling in real-time and in the field. Our modelling workflow is also applicable to creation of virtual worlds, as it supports fast generation of models for real-life objects.

A detailed presentation of our workflow, is included as an accompanying video: http://goo.gl/CeNu .

ACKNOWLEDGMENT

This research is funded through TUBITAK CAREER Grant 105E087.

REFERENCES

[1] Azuma, R., Furmanski, C., oct. 2003. Evaluating label placement for augmented reality view management. pp. 66–75.

[2] Baillot, Y., Brown, D., Julier, S., 2001. Authoring of physical models using mobile computers. In: ISWC ’01: Proceedings of the 5th IEEE International Symposium on Wearable Computers. IEEE Computer Society, p. 39.

[3] Barnes, M., 2006. Collada. In: SIGGRAPH ’06: ACM SIGGRAPH 2006 Courses. ACM, p. 8.

[4] Benko, H., Ishak, E. W., Feiner, S., 2004. Collaborative mixed reality visualization of an archaeological excavation. In: ISMAR ’04: IEEE Computer Society, pp. 132–140.

[5] Blender, 2005. Blender foundation. URL http://www.blender.org

[6] Boissonnat, J.-D., Devillers, O., Teillaud, M., Yvinec, M., 2000. Triangulations in cgal (extended abstract). In: SCG ’00. ACM, New York, NY, USA, pp. 11–18.

[7] Debevec, P. E., Taylor, C. J., Malik, J., 1996. Modeling and rendering architecture from photographs: a hybrid geometry- and image-based approach. In: SIGGRAPH ’96. ACM, pp.11–20.

[8] Feiner, S., MacIntyre, B., Höllerer, T., Webster, A., Oct 1997. A touring machine: prototyping 3d mobile augmented reality systems for exploring the urban environment. In: Wearable Computers, 1997. Digest of Papers., First International Symposium on. pp. 74–81.

[9] Grzeszczuk, R., Kosecka, J., Vedantham, R., Hile, H., 2009. Creating compact architectural models by geo-registering image collections. In: IEEE InternationalWorkshop on 3D Digital Imaging and Modeling.

[10] Harris, E. C., 1989. Principles of archaeological stratigraphy, 2nd Edition. London, New York: Academic Press.

[11] Hartley, R., jun. 1997. In defense of the eight-point algorithm. Pattern Analysis and Machine Intelligence, IEEE Transactions on 19 (6), 580–593.

[12] Henderson, S., Feiner, S., jan. 2010. Opportunistic tangible user interfaces for augmented reality. Visualization and Computer Graphics, IEEE Transactions on 16 (1), 4–16.

[13] Höllerer, T., Feiner, S., Terauchi, T., Rashid, G., Hallaway, D., 1999. Exploring mars: developing indoor and outdoor user interfaces to a mobile augmented reality system. Computers and Graphics 23 (6), 779–785.

[14] Kim, D. H., Kim, M.-J., 2006. A new modeling interface for the pen-input displays. Comput. Aided Des. 38 (3), 210–223.

[15] Kopf, J., Neubert, B., Chen, B., Cohen, M., Cohen-Or, D., Deussen, O., Uyttendaele, M., Lischinski, D., 2008. Deep photo: model-based photograph enhancement and viewing. In: SIGGRAPH Asia ’08: ACM SIGGRAPH Asia 2008 papers. ACM, New York, NY, USA, pp. 1–10.

[16] Luan, Q., Drucker, S. M., Kopf, J., Xu, Y.-Q., Cohen, M. F., 2008. Annotating gigapixel images. In: UIST ’08: ACM, New York, NY, USA, pp. 33–36.

[17] OpenCV, 2009. Open source computer vision. A library of programming functions for real time computer vision. URL www.digilab.uni-hannover.de

[18] PFTRACK, 2010. Thepixelfarm. A commercial camera tracking and image based modelling product. URL http://www.thepixelfarm.co.uk

[19] Piekarski,W., 2006. 3d modeling with the tinmith mobile outdoor augmented reality system. IEEE Comput. Graph. Appl. 26 (1), 14–17.

[20] Piekarski, W., Thomas, B. H., 2003. Interactive augmented reality techniques for construction at a distance of 3d geometry. In: EGVE ’03: ACM, pp. 19–28.

[21] Pollefeys, M., Van Gool, L., Vergauwen, M., Verbiest, F., Cornelis, K., Tops, J., Koch, R., 2004. Visual modeling with a hand-held camera. Int. J. Comput. Vision 59 (3), 207–232.

[22] Rekimoto, J., Nagao, K., 1995. The world through the computer: computer augmented interaction with real world environments. In: UIST ’95: Proceedings of the 8th annual ACM symposium on User interface and software technology. ACM, pp. 29–36.

[23] Schall, G., Mendez, E., Schmalstieg, D., 2008. Virtual redlining for civil engineering in real environments. In: ISMAR ’08. IEEE Computer Society, pp. 95–98.

[24] Sinha, S. N., Steedly, D., Szeliski, R., Agrawala, M., Pollefeys, M., 2008. Interactive 3d architectural modeling from unordered photo collections. In: SIGGRAPH Asia ’08: ACM SIGGRAPH Asia 2008 papers. ACM, pp. 1–10.

[25] Snavely, N., Seitz, S. M., Szeliski, R., 2006. Photo tourism: exploring photo collections in 3d. In: SIGGRAPH ’06: ACM SIGGRAPH 2006 Papers. ACM, pp. 835–846.

[26] Thormählen, T., Broszio, H., 2010. Vodoo camera tracker. URL www.digilab.uni-hannover.de

[27] Thormählen, T., Seidel, H.-P., 2008. 3d-modeling by ortho-image generation from image sequences. In: SIGGRAPH ’08: ACM SIGGRAPH 2008 papers. ACM, pp. 1–5.

[28] van den Hengel, A., jul. 2010. Image-based modelling for augmenting reality. pp. 1–4.

[29] van den Hengel, A., Dick, A., Thormählen, T., Ward, B., Torr, P. H. S., 2007. Videotrace: rapid interactive scene modelling from video. In: SIGGRAPH ’07: ACM SIGGRAPH 2007 papers. ACM, p. 86.

[30] Wither, J., DiVerdi, S., Höllerer, T., 2009. Annotation in outdoor augmented reality. Computers and Graphics 33 (6), 679–689.

[31] Wither, J., Höllerer, T., 2005. Pictorial depth cues for outdoor augmented reality. In: ISWC ’05: IEEE Computer Society, pp. 92–99.

[32] Xiao, J., Fang, T., Tan, P., Zhao, P., Ofek, E., Quan, L., 2008. Image-based façade modeling. In: SIGGRAPH Asia ’08: ACM SIGGRAPH Asia 2008 papers. ACM, pp. 1–10