
ViewFinder: Supporting the Installation and Reconfiguration of Multi-Camera Motion Capture Systems with a Mobile Application

Mehmet Aydın Baytaş

Qualisys AB
Gothenburg, Sweden
Koç University
Istanbul, Turkey
mbaytas@ku.edu.tr

Emmanuel Batis

Chalmers University of
Technology
Gothenburg, Sweden
batis@student.chalmers.se

Mathias Bylund

Chalmers University of
Technology
Gothenburg, Sweden
bylundm@student.chalmers.se

Damla Çay

Koç University
Istanbul, Turkey
dcay13@ku.edu.tr

Asım Evren Yantaç

Koç University
Istanbul, Turkey
eyantac@ku.edu.tr

Morten Fjeld

Chalmers University of
Technology
Gothenburg, Sweden
fjeld@chalmers.se

Abstract

We present ViewFinder, a cross-platform mobile application to support the installation and reconfiguration of marker-based motion capture systems with multiple cameras. ViewFinder addresses a common issue when installing or reconfiguring motion capture systems: that system components such as cameras and the host computer can be physically separate and/or difficult to reach, requiring personnel to maneuver between them frequently and laboriously. ViewFinder allows setup technicians or end-users to visualize the output of each camera in the system in a variety of ways in real time, on a smartphone or tablet, while also providing a means to make adjustments to system parameters such as exposure or marker thresholds on the fly. The app has been designed and evaluated through a process observing user-centered design principles, and effectively reduces the amount of work involved in installing and reconfiguring motion capture systems.

Author Keywords

ViewFinder; motion capture; remote feedback

ACM Classification Keywords

H.5.m [Information interfaces and presentation (e.g., HCI)]:
Miscellaneous

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

Copyright held by the owner/author(s).
MUM 2017, November 26–29, 2017, Stuttgart, Germany
ACM 978-1-4503-5378-6/17/11.
<https://doi.org/10.1145/3152832.3156616>

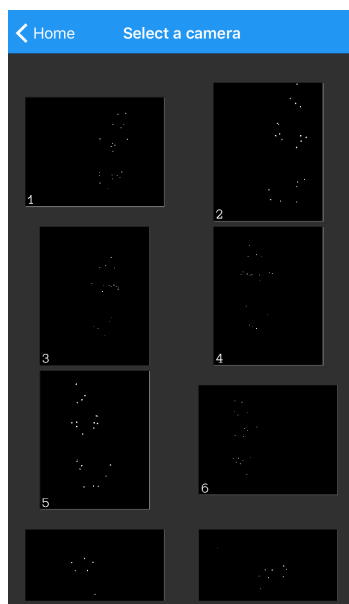


Figure 1: Screenshot from the ViewFinder app, showing an overview of all cameras connected to the motion capture system.

Introduction

Multi-camera motion capture systems are used for position tracking in a wide array of application domains: producing character animations for games and motion pictures; analyzing the technique of athletes, musicians, and other performers in detail; diagnosing neurological and musculoskeletal disorders and providing feedback during physical therapy; tracking multiple bodies for immersive, multi-player virtual reality setups [3], and supporting industrial applications involving the closed-loop control of mechatronic systems. The technology often involves measuring, via photogrammetry, the positions of spherical infrared-reflective markers that are attached to the subject of interest. These markers must remain visible to more than one camera at all times. An occluded or otherwise distorted view of the markers results in errors or 'gaps' in the data, which can require additional labor to correct during post-processing or introduce failures in real-time systems. It is important to ensure that the system can adequately 'see' all of the markers attached to the subjects in the capture volume [1]. Cameras must be positioned correctly, taking expected movements of the markers and the subjects and environmental factors into consideration. Moreover, the environment may be difficult to operate in: cameras may have to be placed underwater or mounted on fixtures on a high ceiling. Thus, setting up a new motion capture system or reconfiguring an existing installation can be a laborious for technicians or end-users, and there is a need for tools that conveniently deliver adequate feedback during the process.

To address the challenge of obtaining feedback about the system configuration while installing and reconfiguring motion capture setups with multiple cameras, we designed and developed ViewFinder, a cross-platform mobile application to visualize the coverage of each camera in a variety of ways, in real time, on a smartphone or tablet. ViewFinder

also allows for its user to view and remotely adjust system parameters such as exposure and thresholding to respond to environmental conditions such as lighting.

ViewFinder was developed through a user-centered design process, aiming to fulfill the needs of diverse user populations which include experienced technicians who perform initial installations at client sites, as well as end-users who may need to tweak an existing system to address changing needs. The following sections describe the final application, the design process, the evaluation of the final prototype, and insights gained throughout the project. ViewFinder is currently deployed in production and can communicate out-of-the-box with camera systems by Qualisys, a Swedish manufacturer; although the application can be adapted to work with other hardware as well. The app can be downloaded by iOS and Android users for free.

The ViewFinder App

To describe how ViewFinder can support the process of installing or reconfiguring a motion capture system, we will consider the case of a technician performing an installation from scratch. Figure 4 depicts this workflow; and throughout this section we refer, with bold numbers in parentheses, to the numbered planes in the figure that break down the various activities involved.

The technician's workflow begins with unpacking the equipment (1), mounting the cameras on their fixtures for a preliminary setup (2), and verifying that the system is functional (3). (The placement of the cameras for the preliminary setup may be determined through convention and experience, as well as using software that enables planning process on-screen or in-situ via augmented reality [2].) The technician then checks the orientations and settings for each camera on the software used to record motion capture



Figure 2: Screenshot from the ViewFinder app showing the detail view with settings for a single camera in the motion capture system.



Figure 3: A Qualisys "Miquis" motion capture camera, where the LED ring around the lens can be remotely controlled by ViewFinder to deliver feedback.

sequences (4) and determines which cameras require fine-tuning. A camera that is designated for fine-tuning in the software must be located in the real world (5). Some cameras may support this task by displaying an identifier on a built-in screen, but many do not, and merely locating a camera for adjustment may become arduous in the absence of appropriate tools. The ViewFinder app, when connected to a Qualisys camera system, supports this task by lighting up the LED rings placed around the camera lens (Figure 3) when the technician navigates to the detail view for a specific camera (Figure 2).

After locating the camera to be adjusted in the real world (6), the technician moves to physically adjust the camera to ensure adequate coverage of the capture volume (7). Without a mobile tool to give feedback on what lies in the camera's field of view, as well as the coverage of the other cameras in the system, this becomes a difficult task. The technician must either look at a distant screen when performing adjustments, which consumes time and compromises safety, or they must recruit additional personnel (8). Using a ViewFinder on a smartphone or tablet, the technician can remotely get feedback on the positioning of every camera connected to the system (Figure 1). After correcting the positioning, since the orientation of a camera can affect the lighting conditions that it faces, camera parameters such as exposure and marker detection thresholds must be adjusted (9, 10). ViewFinder provides options to make these adjustments remotely and update the host system in real time. The completion of these adjustments marks the end of the installation process; the system can now be calibrated, and the measurement session can commence thereafter (11).

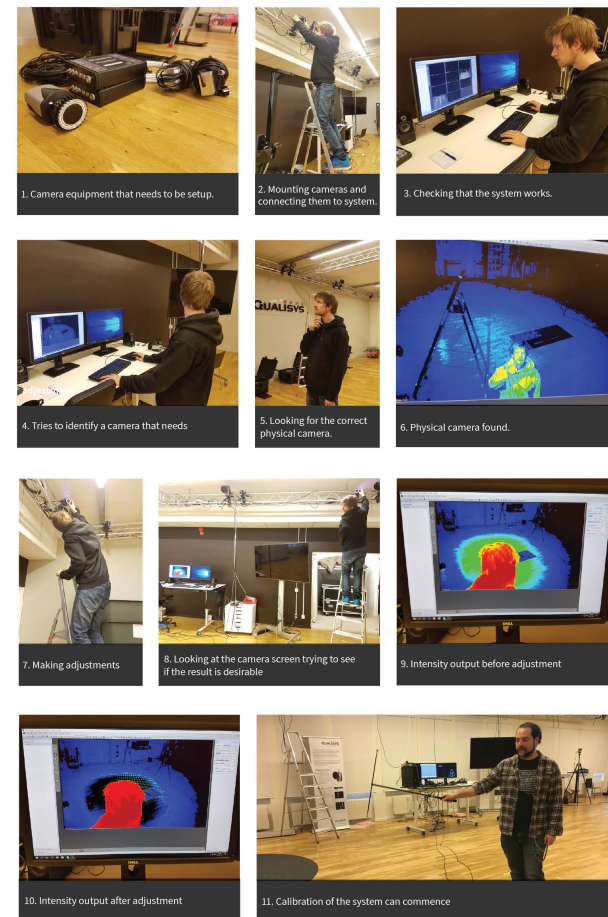


Figure 4: Activities involved in installing a motion capture system.

Implementation Details

ViewFinder has been built using the Xamarin framework for cross-platform mobile applications¹, and runs on commodity smartphones and tablets, with 98% of the codebase (in terms of lines of code) shared across platforms. The open source Qualisys Real-time SDK² is used for networking between the app and motion capture systems over Wi-Fi. The use of a cross-platform framework and open-source components were important maintainability-related considerations that emerged from early requirements gathering.

Design and Evaluation

We have designed and evaluated ViewFinder through a process observing user-centered design principles. We began the process by attempting to understand the experience of setting up a motion capture system, from the perspectives of technicians and end-users. To this end, loosely following the framework of *contextual task analysis* [6], we experimented with collecting motion capture data to acquire first-hand domain knowledge, and conducted observation sessions and interviews with potential users. For these sessions, as well as for subsequent refinement, feedback gathering, and design evaluation, we recruited 4 engineers who worked at a motion capture systems vendor and performed installations (as well as system development) as part of their job, and 2 researchers who used motion capture systems to collect data from subjects. We then articulated our findings from this process in two forms: *scenarios* (see Table 1 and *requirements* (which were further broken down as pertaining to functional, data, environmental, user, and usability aspects of the design). The requirements generated at this stage were manifold, and they were pruned through a process based on the *KJ-Method*, also known as affinity

¹xamarin.com

²github.com/qualisys/RTClientSDK.Net

Scenario	Description
Ladder	The user climbs up a ladder to move a camera into position. (Before making adjustments to the camera position the user has to place the mobile device somewhere.) During adjustments the user will repeatedly look at the device and with it when needed.
Rotation	The user needs to physically rotate the camera into a 'portrait' orientation for functional purposes. The user then needs to rotate the camera view in the application to the correct orientation.
Identification	The user needs to identify a specific camera in the room that corresponds with one of the camera windows in the application.
Focus	The user needs to adjust the focus of a camera, and has to be able to view the 'intensity' of a marker sphere in the camera's view to make adjustments accordingly.

Table 1: Summaries for four concise scenarios describing how ViewFinder will be used while installing or reconfiguring a motion capture system.

diagramming [10], before proceeding to prototyping application designs.

Based on the requirements elicited through the process described above, we first produced modular paper prototypes of user interface ideas and gathered feedback on various design elements from the engineers and end-users. We then implemented interactive wireframes to collect feedback on the navigational flow, as well as on higher-fidelity concepts in the user interface design. Through a semi-structured, iterative cycle, we conferred twice with each researcher in our user cohort, and 2-3 times with each engineer, before finalizing the design and implementing the working application.

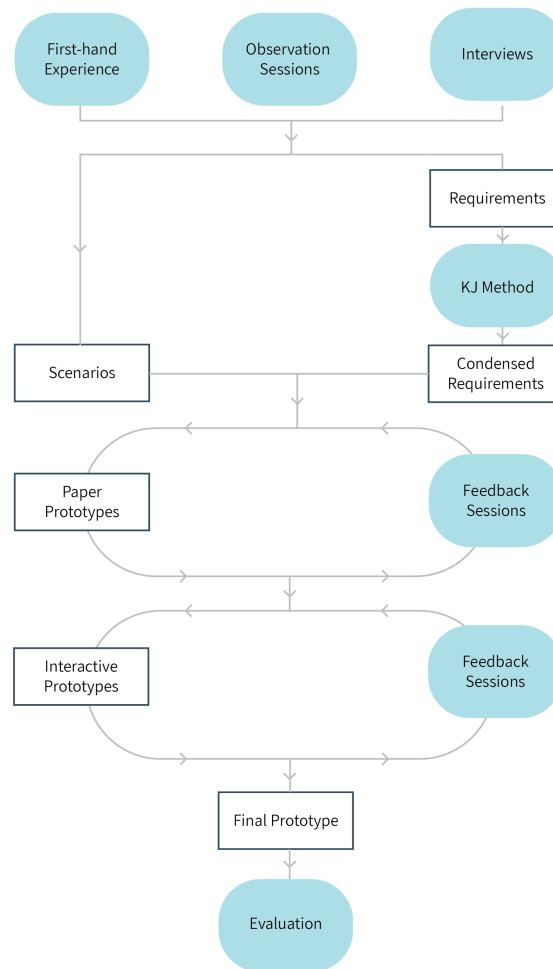


Figure 5: An overview of the design process.

In addition to intermediate, informal evaluation steps during development, a preliminary evaluation of the final prototype was carried out with 6 participants, representing different user archetypes. Three of the participants were experience motion capture specialists or developers, two were researchers who collected and worked with motion capture data regularly, and one was a software developer who had little experience with the motion capture system itself. Here, we aimed to assess how ViewFinder supports the work involved in setting up or reconfiguring a motion capture system, and to evaluate the usability of the application. First, short studies were conducted to assess the usability of specific key features in the application. We provided a running instance of ViewFinder to participants, and allowed them to familiarize themselves with the application for five minutes. Users were asked to complete 4 different tasks, and subsequently answer a Single Ease Question (SEQ) regarding their experiences in performing each task. We chose to use the SEQ because it is a quick, efficient method that correlates with more fine-grained usability measures [8, 9, 7]. The tasks, based on the use cases elicited during the earlier stages of the design process, are given in Table 2, together with the mean results from the answers to the SEQ. We also asked the same participants to go through workflows based on the four scenarios articulated beforehand, recorded how they made use of the application to support their work as they reconfigured part of a 4-camera motion capture system, and interviewed them afterwards to obtain feedback.

The results from the preliminary study indicate that overall, participants were able to perform the atomic tasks with ease, which implies evidence for the usability of the application design. Observations and feedback from the sessions where participants were asked to go through the short application scenarios show that the five experienced motion

#	Task Description	SEQ
1	Navigate to view Camera 2 only.	5.5
2	Ensure that Camera 4 is running in 'intensity mode.'	4
3	Switch every camera to 'marker mode,' except for Camera 4.	5.7
4	Lower the marker detection threshold for Camera 1.	6.7

Table 2: Tasks used for short user studies, based on use cases elicited during the design process. The column "SEQ" gives the mean of the SEQ scores for each task, collected from all 6 participants.

capture users were able to integrate ViewFinder into their work with ease, and they were vocal about their appreciation for having access to mobile feedback. The novice user, who had limited (although not non-existent) experience with the motion capture setup itself, was able to complete three of the scenarios easily, but struggled on the 'Focus' case because they were unfamiliar with the core features of the system. Overall, our findings show that ViewFinder is successful in supporting the work involved in setting up or reconfiguring a motion capture system and cutting the amount of time and effort required to perform related tasks.

Conclusion

We presented the design, implementation, and evaluation of ViewFinder, a cross-platform mobile application that supports the installation and reconfiguration of marker-based motion capture systems with multiple cameras, by providing real-time visual feedback on a commodity smartphone or tablet. We discussed the design process and a preliminary evaluation with the final prototype of the application, which were conducted in a manner that observes princi-

ples of user-centered design. A preliminary evaluation with six users has shown that the application is able to provide mobile feedback to support the tasks involved in setting up motion capture systems, and that the interface design is appropriate for users with different levels of experience with the motion capture hardware; and experienced/specialist users have appreciated the availability of mobile feedback, which addressed pain points with their existing tools and workflows. Future work can investigate encapsulating and communicating domain knowledge and best practices in the application to further support novice users, and extending the use cases for the mobile application toward other stages of the motion capture process, such as calibration [4] or the measurement itself [5].

Acknowledgements

This work is part of a project that has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 676063.

REFERENCES

1. B. Danette Allen and Greg Welch. 2005. A General Method for Comparing the Expected Performance of Tracking and Motion Capture Systems. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology (VRST '05)*. ACM, New York, NY, USA, 201–210. DOI: <http://dx.doi.org/10.1145/1101616.1101658>
2. Mehmet Aydın Baytaş, Asım Evren Yantaç, and Morten Fjeld. 2017. LabDesignAR: Configuring Multi-Camera Motion Capture Systems in Augmented Reality. In *Proceedings of the 23rd ACM Symposium on Virtual Reality Software and Technology (VRST '17)*. ACM, New York, NY, USA. DOI: <http://dx.doi.org/10.1145/3139131.3141778>

3. Noah Bloem. 2017. Zombies?! Surviving the undead legion at the VR Arcade Amsterdam. *DutchReview* (5 2017). Retrieved August 17, 2017 from <https://dutchreview.com/dutch-reviews/places/zombies-surviving-the-undead-legion-at-vr-arcade-amsterdam/>.
4. Zlatko Franjic, Paweł W. Woźniak, Gabriele Kasparavičiūte, and Morten Fjeld. 2016. WAVI: Improving Motion Capture Calibration Using Haptic and Visual Feedback. In *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '16)*. ACM, New York, NY, USA, 254–265. DOI: <http://dx.doi.org/10.1145/2935334.2935374>
5. Daniel Kade, Kaan Akşit, Hakan Ürey, and Oğuzhan Özcan. 2015. Head-mounted Mixed Reality Projection Display for Games Production and Entertainment. *Personal Ubiquitous Comput.* 19, 3-4 (July 2015), 509–521. DOI: <http://dx.doi.org/10.1007/s00779-015-0847-y>
6. Deborah J. Mayhew. 2008. Requirements Specifications within the Usability Engineering Lifecycle. In *The Human-Computer Interaction Handbook*, Andrew Sears and Julie Jacko (Eds.). Lawrence Erlbaum Associates, New York, NY, USA, Chapter 47, 917–926.
7. Jeff Sauro. 2012. 10 Things To Know About The Single Ease Question (SEQ). (2012). <https://measuringu.com/seq10/>.
8. Jeff Sauro and Joseph S. Dumas. 2009. Comparison of Three One-question, Post-task Usability Questionnaires. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '09)*. ACM, New York, NY, USA, 1599–1608. DOI: <http://dx.doi.org/10.1145/1518701.1518946>
9. Jeff Sauro and James R. Lewis. 2009. Correlations Among Prototypical Usability Metrics: Evidence for the Construct of Usability. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '09)*. ACM, New York, NY, USA, 1609–1618. DOI: <http://dx.doi.org/10.1145/1518701.1518947>
10. Jared M. Spool. 2004. The KJ-Technique: A Group Process for Establishing Priorities. (2004). https://articles.uie.com/kj_technique/.