Distributed Heterogeneous Tracking for Augmented Reality

Mihran Tuceryan
Indiana University Purdue University Indianapolis, USA

Rajeev R. Raje
Indiana University Purdue University Indianapolis, USA

INTRODUCTION

Augmented reality (AR) is a technique in which a user's view of the real world is enhanced or augmented with additional information generated from a computer model (Azuma et al., 2001). The enhancement may consist of virtual artifacts to be fitted into the environment or a display of non-geometric information about existing real objects. Mobile AR (MAR) systems implement this interaction paradigm in an environment in which the user moves, possibly over wide areas (Feiner, MacIntyre, Hoellerer, & Webster, 1997). This is in contrast to non-mobile AR systems that are utilized in limited spaces such as a computer-aided surgery or by a technician's aid in a repair shop. There are a number of challenges to implementing successful AR systems. These include a proper calibration of the optical properties of cameras and display systems (Tuceryan et al., 1995; Tuceryan, Genc, & Navab, 2002), and an accurate registration of three-dimensional objects with their physical counterparts and environments (Breen, Whitaker, Rose, & Tuceryan, 1996; Whitaker, Crampton, Breen, Tuceryan, & Rose, 1995). In particular, as the observer (or an object of interest) moves over time, the 3D graphics need to be properly updated so that the realism of the resulting scene and/or alignment of necessary objects and graphics are maintained. Furthermore, this has to be done in real time and with high accuracy. The technology that allows this real-time update of the graphics as users and objects move is a *tracking system* that measures the position and orientation of the tracked objects (Koller et al., 1997). The ability to track objects, therefore, is one of the big challenges in MAR systems. This article describes a software framework for realizing such a distributed tracking environment by discovering independently deployed, possibly heterogeneous trackers and fusing the data from them while roaming over a wide area. In addition to the MAR domain, this kind of a tracking capability would also be useful in other domains such as robotics and location-aware applications. The novelty of this research lies in the amalgamation of the theoretical principles from the domains of AR/VR, data fusion, and the distributed software systems to create a sensor-based, wide-area tracking environment.

BACKGROUND

Although a few approaches for tracking have been proposed (e.g., Hightower & Borriello, 2001; Koller et al., 1997; Neumann & Cho, 1996; State, Hirota, Chen, Garrett, & Livingston, 1996), the ability to track objects accurately and in real time over a *wide area* does not yet have a satisfactory solution. Moreover, many of these approaches require a highly engineered environment with a uniform set of trackers whose architecture is known in advance (Welch, & Foxlin, 2002; Ubisense, 2006). Assuming that trackers have been deployed and are operating and exist in the environment, this research deals with questions of how to discover what trackers exist in a local area, what quality-of-service (QoS) properties they have, and how to make the best use of their measurements in a mobile and dynamic environment.

The wide-area, ubiquitous tracking that is the focus of this article has been addressed mainly in the pervasive/ubiquitous computing community. An early tracking system was HiBall that utilized a ceiling instrumented by LED lights (Welch, 1999). The HiBall tracker covered a wider area than a typical magnetic tracking system, and in the implementation its range covered a room or a lab. The scalability of such a system was limited because of the increased cost of extending beyond the size of a lab. The BAT system, which used ultrasound as the core technology (Harter, Hopper, Steggles, Ward, & Webster, 2002; Newman, Ingram, & Hopper, 2001), had a limited resolution. The location sensing system, by Ubisense (2006) uses the ultra-wide-band technology and has a better resolution (6 inches positional accuracy, according to company Web sites).

Researchers at Intel Research studied the use of existing wireless hotspots and cell phone towers to compute location information over wide-areas (Schilit et al., 2003; Borriello, Chalmers, LaMarca, & Nixon, 2005). Bahl and others studied localization techniques using existing Wi-Fi wireless hubs (Bahl & Padmanabhan, 2000; Balachandran, Voelker, & Bahl, 2003). Their methods assume a ubiquitous infrastructure that exists for other purposes (networking) that can be tapped into for localization of users. Typically, their resolution tends to be low and is not sufficient for typical AR applications.

Ubiquitous tracking systems specifically for AR systems have been also studied by Bauer et al. (2002). Newman et al. (2004), and Reitmayr (2001), and have resulted in prototypical systems, some of which are component based (e.g., DWARF by [Bauer, Bruegge, Klinker, MacWilliams, Reicher, Riss, et al., 2001]).

THE UNIFRAME-BASED MOBILE TRACKING SERVICE FOR AR

As indicated earlier, the distributed tracking system is an example of a heterogeneous, distributed computing system. The overall architecture and various components of the distributed tracker subsystem that is the focus of this article are shown in Figure 1.

The software realization of this tracking system is based on the principles of uniframe (Olson, Raje, Bryant, Burt, & Auguston, 2005). Uniframe provides an environment for an interoperation of heterogeneous and distributed software components, and uses the principles a meta-component model, service-oriented architectures, generative programming, and two-level (TLG) and event grammars (EG). The realization of the distributed tracking system, using the UniFrame, begins with a generative domain model (GDM) (Czarnecki & Eisenecker, 2000) created by experts from the tracking domain. This GDM contains various details, such as the software architecture of the tracking system expressed in terms of underlying components, their interactions, the rules for generating middleware, and the rules for the prediction and monitoring of the quality of the integrated system. Each component is defined by a Unified Meta-component Model (UMM) specification (Raje, 2000). The UMM has three parts: (a) components; (b) services, offered by components, and associated guarantees; and (c) infrastructure for deploying and discovering components. A developer who wishes to create specific components for the tracking system consults the GDM and creates implementations using the UMM specifications encoded there. After a component is developed, it is validated against the quality requirements, both functional and QoS. The developer also creates an associated UMM specification for that component. This specification and the component are deployed on the network. These components are also registered with the uniframe resource discovery system (URDS) (Siram, 2002).

A system integrator planning to create the tracking system, from independently developed and deployed components, issues a query consisting of the requirements the tracking system must meet. The query processing consults the GDM, divides the query into many sub-queries, each corresponding to a single component UMM specification. These sub-queries are passed to the URDS, which searches for appropriate matching components. If such components are found, these are returned to the system integrator, who then selects a subset of these results, provides any proprietary components, and requests the process to assemble the integrated system conforming to the design. The uniframe system generator (Huang, 2003) carries out the generation of the integrated system. The key challenges in creating the tracking system are: (a) designing the GDM, (b) the discovery of components, and (c) the generation of a prototypical tracking system. These are briefly discussed below.

Designing the GDM

The GDM is developed by the domain experts and contains the software architecture of the family of systems, along with many associated details. For a tracking system, it can be either handcrafted or generated via the uniframe system generator (Huang, 2003). One important piece of the GDM relates to the specification of components that make the software architecture of the tracking system. The specification provides

Figure 1. Architecture and components of the distributed tracking subsystem for mobile AR applications
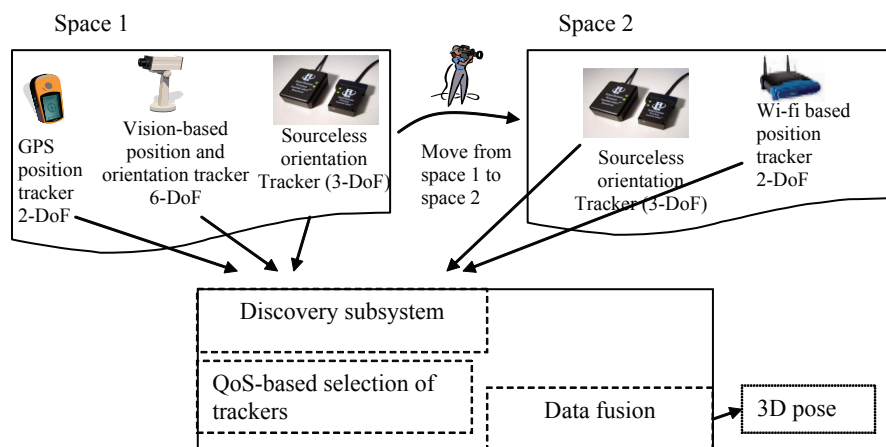
Related Content

Order Statistics and Applications
E. Jack Chen (2019). *Advanced Methodologies and Technologies in Network Architecture, Mobile Computing, and Data Analytics (pp. 440-457).*
www.irma-international.org/chapter/order-statistics-and-applications/214634

Online Teaching System of Sports Training Based on Mobile Multimedia Communication Platform
Tianxiang Yueand Yebing Zou (2019). *International Journal of Mobile Computing and Multimedia Communications (pp. 32-48).*
www.irma-international.org/article/online-teaching-system-of-sports-training-based-on-mobile-multimedia-communication-platform/220421

Quality of Service Issues in Mobile Multimedia Transmission
Nalin Sharda (2009). *Mobile Computing: Concepts, Methodologies, Tools, and Applications (pp. 2843-2861).*
www.irma-international.org/chapter/quality-service-issues-mobile-multimedia/26697

Context as a Necessity in Mobile Applications
Eleni Christopoulou (2008). *Handbook of Research on User Interface Design and Evaluation for Mobile Technology (pp. 187-204).*
www.irma-international.org/chapter/context-necessity-mobile-applications/21831

Modelling and Simulation of Mobile Mixed Systems
Emmanuel Dubois, Wafaa Abou Moussa, Cédric Bachand Nelly de Bonnefoy (2008). *Handbook of Research on User Interface Design and Evaluation for Mobile Technology (pp. 346-363).*
www.irma-international.org/chapter/modelling-simulation-mobile-mixed-systems/21841