# History and Future of Tracking for Mobile Phone Augmented Reality

Daniel Wagner and Dieter Schmalstieg
*Graz University of Technology*
*{ wagner, schmalstieg } @icg.tugraz.at*

## Abstract

*We present an overview on the history of tracking for mobile phone Augmented Reality. We present popular approaches using marker tracking, natural feature tracking or offloading to nearby servers. We then outline likely future work.*

## 1. Introduction

Augmented Reality (AR) and Virtual Reality (VR) require 6DOF pose tracking of devices such as head-mounted displays, tangible interface objects, etc. Pose tracking must be inexpensive, work robustly and in real time in changing environmental conditions. Additionally it should support a large working volume and provide automatic localization in global coordinates. However, a guaranteed level of accuracy is usually not required. In this paper we give an overview on the history of tracking for Augmented Reality on mobile phones, which are attractive for end users due to their low costs and wide spread.

Before the advent of mobile phone AR, some researchers working on mobile Augmented Reality (AR) had started replacing the cumbersome backpack plus head-mounted display setups (see Figure 1a) with ultra mobile PCs (UMPCs, Figure 1b). PDAs (Figure 1c) are kind of a predecessor of smartphones (Figure 1d). These two platforms have merged into a single device class and PDAs have vanished from the market.

Compared to more powerful and larger UMPCs, smartphones are aiming for a different market. Price, weight and battery life are designed for a large consumer base and mobile - rather than merely portable - operation. Unmodified consumer devices are also surprisingly robust and foolproof given their fragile appearance. However, these desirable properties come at the price of restricted computing capabilities compared to the PC platform. Achieving sufficient performance for AR applications requires careful choice of algorithms and optimized code.

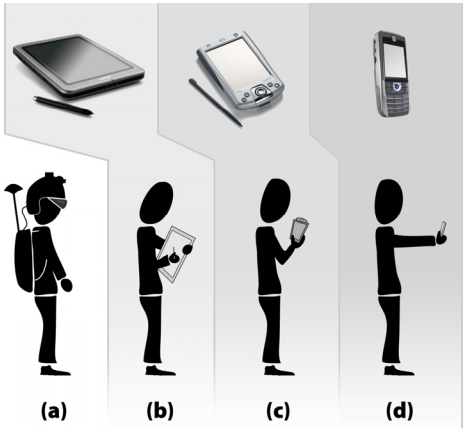


**Figure 1: The evolution and miniaturization of mobile AR: (a) Backpack with HMD, (b) UMPC, (c) handheld, (d) Mobile phone**

Any self-contained mobile AR setup should be capable of determining its own pose using its own sensors. The built-in camera available on most mobile devices naturally lends itself to computer vision approaches. However, the quality of computer vision tracking is strongly influenced by camera and image sensor characteristics, such as frame size, update rate, color depth or lens distortion, which tend to be rather poor on low-end devices. Unfortunately marketing has driven the development to more megapixels rather than higher video quality.

The combination with other sensors, such as inertial measurement units or GPS, can dramatically enhance the capabilities of handheld tracking. Until recently only few such devices were equipped with multiple sensors, which is why most of the work presented in the following uses camera-based tracking only.

A major obstacle for AR tracking on mobile phones comes from the limited processing capabilities of these devices. Typical clock rates are between 200 and 600 MHz, single core and no floating point unit. It is interesting to notice that over the last 5 years the available computational resources have improved by

978-0-7695-3704-7/09 $25.00 © 2009 IEEE
DOI 10.1109/ISUVR.2009.11

only about 100%. The reason for this stagnation is that battery power is the main constraint for mobile phone design and has improved only ~10% per year.

## 2. Tracking by Outsourcing to a PC

One approach to overcome the resource constraints of mobile devices is to outsource tracking to PCs connected via a wireless connection. All of these approaches suffer from low performance due to restricted bandwidth. Additionally the imposed infrastructure dependency limits scalability in the number of client devices as well as robustness of the system, which fails to work if no wireless connection is available.

The AR-PDA project [2] used digital image streaming from and to an application server. All processing tasks, including tracking, rendering and application logic were outsourced degrading the client device to a pure display plus camera.

Shibata's work [12] was an extension of this concept and allowed to adapt how much work it outsourced. The project aimed at load balancing between client and server - the weaker the client, the more tasks are outsourced to a server. In a more recent project Hile et al. report a SIFT based indoor navigation system [6], which relies on a server to do all computer vision work. Zöllner et. al developed a tracking system [20] that combines Randimized Trees for detection with KLT for tracking. The system runs in real-time on a PC. A mobile client can send pictures via a wireless connection.

The server-based approaches mentioned above are not real-time. Typical response times are reported to be ~10 seconds for processing a single frame. When using wide area connectivity, such as GSM or UMTS, connecting to the server and uploading image data creates a delay of several seconds thereby preventing interactive frame rates.

## 3. Marker Tracking

Naturally, first inroads in tracking on mobile devices themselves focused on fiducial marker tracking, which is less computationally demanding than approaches based on natural features. Nevertheless, only few solutions for mobile phones have been reported in literature. In 2003 Wagner et. al ported ARToolKit to Windows CE and thus created the first self-contained AR application [15] on an off-the-shelf embedded device. This port later evolved into the ARToolKitPlus library [16].

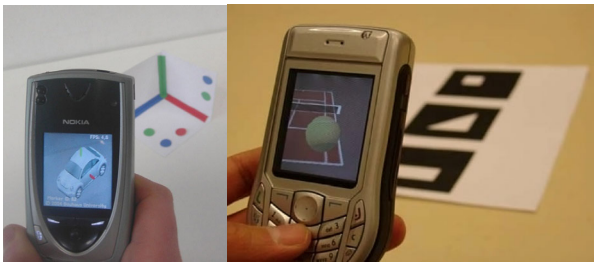

**Figure 2: Left: 3D markers by Möhring. Right: AR-Tennis by Henrysson.**

In 2004 Möhring [9] created a tracker for mobile phones that detects color-coded 3D marker shapes (see Figure 2, left). The system's accuracy was very limited, since it did not take camera calibration or sub-pixel accuracy into account.

In 2005 Henrysson [5] created a Symbian port of ARToolKit, partially based on the ARToolKitPlus source code, which was used for the AR Tennis game (see Figure 2, right), the first 2-player AR game on mobile phones.

Around the same time Rohs created the VisualCodes system for Symbian smartphones [11]. Similar to Möhring's approach, VisualCodes (see left image in Figure 3) provides only simple tracking of 2D position on the screen, 1D rotation and a very coarse distance measure.

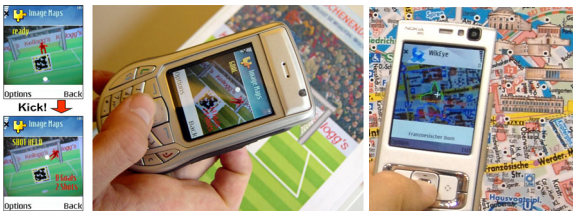


**Figure 3: Left: Visual Codes by Rohs. Right: WikEye by Hecht et al.**

A similar tracking quality is delivered by the approach of the WikEye project [4] that tracks and augments maps that are overlaid with a regular grid of dots (see right image in Figure 3).

In 2008 Wagner et al. created Studierstube Tracker [17], a marker library supporting many different types of markers on mobile phones.

## 4. Natural Feature Tracking

So far, there has been very little work on natural feature tracking for Augmented Reality on mobile phones. Wagner et al. use an approach loosely related to SIFT [8] and Ferns [10] to create the first real-time

6DOF natural feature tracking system running on mobile phones [18]. Their system is able to detect and track a small number of planar objects (~20) at interactive frame rates on average smart phones. Recent work by Ta et al. [13] uses a more computationally intensive but also more accurate tracking directly in feature scale space.

## 5. Non-AR Approaches

There have been several approaches to tracking on mobile phones that are not sufficient for Augmented Reality, but still deserve mentioning.

One of the first AR-like applications on a mobile phone was the Mosquito Hunt game (see Figure 4) on the Siemens SX-1 phone in 2003. It used optical flow detection to estimate the movement of the rotational mobile phone and let the user aim and shot at virtual mosquitoes. The AR soccer game [3] used a similar approach to detect movements of the player's foot trying to shot a virtual ball into a virtual goal.

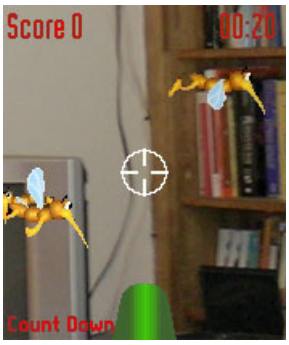


**Figure 4. Mosquito Hunt on the Siemens SX-1.**

TinyMotion [19] by Wang et al. is an open source library for real-time optical flow tracking on mobile phones, but does not deliver any kind of pose estimation. More recently, Takacs et al. created a full SURF [1] implementation for mobile phones [14]. They do not target real-time 6DOF pose estimation, but maximum detection quality and are able to detect a mobile user's coarse position from camera images of facades.

## 6. Future of Tracking on Mobile Phones

The most important recent improvement in hardware is the introduction of hardware floating point support, but even today only a few high end smartphones possess this attractive capability.

Compared to a high end mobile phone, a fast PC (quad-core, 3GHz) has a speed advantage of ~3000%. Even with the upcoming introduction of higher clock rates and multi-cores for mobile phones, this gap will remain significant. Additionally, PCs will also improve in speed so that the performance gap will remain in that order for a foreseeable time.

Instead, we expect additional processing units to become attractive for computer vision tasks. Some of the latest mobile phones possess GPUs with vertex and fragment shaders that are freely programmable. Although not as flexible as CPUs, these GPUs can provide tremendous performance gains for simple tasks. Additionally, an increasing number of mobile phones are equipped with digital signal processors (DSPs) that are optimized to work with small local datasets highly and are therefore suitable for tasks such as image processing. Until now these DSPs are not available to 3rd party programmers, but we expect this to change in the near future.

Almost any mobile phone AR application today uses the mobile phone as a portable rather than mobile device. This is mostly the case due to weaknesses in the currently available tracking systems. Latest research has shown that natural feature tracking is viable on mobile phones, but so far these approaches are limited to small areas only.

To truly take advantage of the phones' mobility, wide area pose tracking systems need to be developed. Such tracking systems must be able to locate the user in a large environment, such as a building or city. Naturally a tracking model of this size will hardly fit into a mobile phone's memory. More importantly though it is usually more practical to build, extend and maintain such a model on a server system that then provides localization services to mobile users.

In such a scenario, a mobile device can send a camera image plus additional information such as GPS position or cell tower IDs to the localization service. The server can then reply by transmitting the device's pose plus tracking data for the close proximity, allowing the mobile phone to track its pose without further external help.

The system described above is the basis for a new concept called Augmented Reality 2.0. Similar to Web 2.0, AR 2.0 strongly involves users to create and extend a shared virtual space. In contrast to traditional AR applications, which are based on prepared content, AR 2.0 invites users to create new content in-situ and share it with other users. Additionally to a tracking service, an infrastructure for storing and distributing virtual content needs to be developed and deployed.

All AR techniques mentioned so far involve a previously created tracking model. However, in some scenarios a model that provides an absolute pose is not required or hard to build and maintain. Recently,

simultaneous localization and mapping (SLAM), a technique originating from robotic research, has beed successfully applied to AR systems [7]. A SLAM tracker creates a 3D environment model on the fly using epipolar geometry and is therefore able to provide high quality tracking in previously unknown environments. However, so far SLAM systems are too computationally demanding for mobile phones and no systems running on this device class have been demonstrated yet.

## 7. Conclusions

The improvements in AR tracking over the last 5 years are mainly possible due to refined methods rather more powerful hardware. Since mobile phone design is strongly driven by battery power, we expect this trend to continue for at least several more years. In the mid term additional units such as GPUs and DSPs will provide more processing power. Most importantly though, it will continue to be necessary simplify and tune complex methods for suitability on mobile phones.

## 8. Acknowledgements

This research was sponsored in part by the Austrian Science Fund FWF under contract Y193, the EU project IPCity (FP6-2004-IST-4-27571) and the Christian Doppler Lab for Handheld Augmented Reality.

## 9. References

[1] Bay, H., Tuytelaars, T., Gool, L. V., Surf: Speeded up robust features, In Proc. ECCV 2006, 2006

[2] Gausemeier, J., Fruend, J., Matysczok, C., Bruederlin, B., Beier, D., Development of a real time image based object recognition method for mobile AR-devices, Proc. of the 2nd International Conference on Computer Graphics, Virtual Reality, Visualisation and Interaction in Africa (Afrigraph 2003), pp. 133-139, 2003

[3] Geiger, C., Paelke, V., Reimann, C.: Mobile Entertainment Computing, In Lecture Notes in Computer Science, Vol. 3105 / 2004, pp. 142-147, 2004

[4] Hecht, B., Rohs, M., Schöning, J., Krüger, A., Wikeye - Using Magic Lenses to Explore Georeferenced Wikipedia Content, In Proc. of the 3rd International Workshop on Pervasive Mobile Interaction Devices (PERMID), 2007

[5] Henrysson, A., Billinghurst, M, Ollila, M.. Face to Face Collabora-tive AR on Mobile Phones. Proceedings International Symposium on Augmented and Mixed Reality (ISMAR'05), pp. 80-89, 2005

[6] Hile, H., Borriello, G., Information Overlay for Camera Phones in Indoor Environments, 3rd International Symposium on Location- and Context-Awareness (LoCA 2007), pp. 68-84, 2007

[7] Klein G, Murray D. Parallel Tracking and Mapping for Small AR Workspaces. In Proc. International Symposium on Mixed and Augmented Reality (ISMAR'07, Nara)

[8] Lowe, D., Distinctive image features from scale-invariant keypoints. Int. Journal of Computer Vision, Volume 60, Issue 2, pp. 91-110, 2004

[9] Möhring, M., Lessig, C., Bimber, C., Video See-Through AR on Consumer Cell Phones. Proceedings of International Symposium on Augmented and Mixed Reality (ISMAR'04), pp. 252-253, 2004

[10] Ozuysal, M., Fua, P., Lepetit, V., Fast keypoint recognition in ten lines of code. In of Proc. CVPR 2007, pp. 1-8, 2007

[11] Rohs, M., Gfeller, B., Using Camera-Equipped Mobile Phones for Interacting with Real-World Objects. Advances in Pervasive Computing, Austrian Computer Society (OCG), pp. 265-271, 2004

[12] Shibata, F., Mobile Computing Laboratory, Department of Computer Science, Ritsumeikan University, Japan, http://www.mclab.ics.ritsumei.ac.jp/research.html

[13] Ta D., Chen W., Gelfand N., Pulli K. SURFTrac: Efficient Tracking and Continuous Object Recognition using Local. To appear in: Proc. IEEE CVPR, 2009.

[14] Takacs, G., Chandrasekhar, V., Gelfand, N., Xiong, Y., Chen, W.-C., Bismpigiannis, T., Grzeszczuk, R., Pulli, K., and Girod, B., Outdoors Augmented Reality on Mobile Phone using Loxel-Based Visual Feature Organization, In Proc. of the 1st ACM international conference on Multimedia information retrieval, pp. 427-434, 2008

[15] Wagner, D., Schmalstieg, D., First Steps Towards Handheld Augmented Reality. Proceedings of the 7th International Conference on Wearable Computers (ISWC 2003), pp. 127-135, 2003

[16] Wagner, D., Schmalstieg, D., ARToolKitPlus for Pose Tracking on Mobile Devices, Proceedings of 12th Computer Vision Winter Workshop (CVWW'07), pp. 139-146, 2007

[17] Wagner, D., Langlotz, T., Schmalstieg, D., Robust und Unobtrusive Marker Tracking on Mobile Phones, In Proc. 7th IEEE/ACM International Symposium on Mixed and Augmented Reality, (ISMAR'08), pp. 121-124, 2008

[18] Wagner, D., Reitmayr, G., Mulloni, A., Drummond, T., Schmalstieg, D., Pose Tracking from Natural Features on Mobile Phones, In Proc. of ISMAR 2008, pp. 125-134, 2008

[19] Wang, J. Zhai, S., Canny, J., Camera Phone Based Motion Sensing: Interaction Techniques, Applications and Performance Study, In ACM UIST 2006, pp. 101-110, 2006

[20] Zoellner, M., Pagani, A., Wüst, H., Stricker, D., Pastarmov, Y., Reality Filtering: A Visual Time Machine in AR, In Proc. of VAST 2008, 9th International Symposium on Virtual Reality, Archaeology, and Intelligent Cultural Heritage, 2008