

Mobile Augmented Reality Support for Architects based on Feature Tracking Techniques

Michael Bang Nielsen¹, Gunnar Kramp², and Kaj Grønbæk¹

¹Department of Computer Science, Aarhus University

Åbogade 34, 8200 Århus N, Denmark.

E-mail: {bang, kgronbak}@daimi.au.dk

²Aarhus School of Architecture

Nørreport, 8000 Århus, Denmark.

E-mail: gunnar.kramp@a-aarhus.dk

Abstract. This paper presents a mobile Augmented Reality (AR) system supporting architects in visualizing 3D models in real-time on site. We describe how vision based feature tracking techniques can help architects making decisions on site concerning visual impact assessment. The AR system consists only of a tablet PC, a web cam and a custom software component. No preparation of the site is required and accurate initialization, which has not been addressed by previous papers on real-time feature tracking, is achieved by a combination of automatic and manual techniques.

1. Introduction

The AR system described in this paper has been developed in the WorkSPACE project, which is an EU IST project focusing on developing interactive working environments for architects. Among other things the project has developed a 3D system called Topos, which is collaborative spatial environment for sharing documents and 3D models. Topos has been extended with various kinds of Augmented Reality components to support integration of the architects' documents and models into the real physical world.

Augmented Reality research [1][11][12], focus on linking digital information and virtual objects to physical objects and places. Typical applications are to link and display digital annotations on top of objects and places by means of some identifying code. Examples are Cybercodes [13] and ARToolkit [2] allowing information to be linked to objects visually tagged with a two-dimensional bar-code label, which is interpreted by a camera-based reader. Several mobile augmented reality systems exist. The MARS prototype [8] is an example of an AR system applying very precise GPS and see-through glasses to provide historical information superimposed onto a campus visitor's view of Columbia University's campus. Another example is the use of a Rekimoto's CyberCode-based Navicam, which recognizes visual tags and superimposes digital text, images, and 3D models on top of the visual tag (the CyberCode). A final example is the Archeoguide system [15], which is a vision based system geared to mix models of old buildings and findings into real-time streams of

video of the archeological site. The system is based on registration of reference photos in fixed positions.

Our AR system called the SitePack is a mobile system combining GPS tracking for location based photo capture and annotation and vision based feature tracking for real-time visualizations.

Section 2 describes the design and implementation of the SitePack software. Section 3 shows how the SitePack was used by landscape architects in evaluating a design proposal on a building site. Section 4 compares the SitePack system to similar systems. Finally, Section 5 concludes the paper.

2. The SitePack Software

On site, architects need to make assessments of the visual impact of their design proposals. This can be facilitated by overlaying the architects' 3D model on a real-time video stream on site. Our mobile AR system, the SitePack, is depicted in figure 1. It runs the Topos 3D environment that supports a number of tracking technologies. In [5] we discussed how GPS and digital compass are sufficiently accurate when supporting collaboration, collecting materials and doing annotations. Here we describe the vision based feature tracker we use for on-site real-time visualizations.



Figure 1: The SitePack appliance is assembled from off-the-shelf hardware components, consisting of a tablet PC with a pen input device, 1GHZ processor, 512 MB RAM, and a 16MB graphics card. The external devices are a 640 x 480 pixel web cam integrated in a bracket mount, and a GPS with digital compass.

2.1. Requirements from the Architectural Domain

The challenges inherent in supporting the visual impact assessment task of architects imply a number of key requirements for the tracking software. In particular it must:

- be robust to a dynamic environment, e.g. occlusions and changing weather.
- require minimal preparation of and intrusion on the environment.
- not be subject to significant drift (i.e. where the 3D overlay drifts from its true position over time), as the system may be in use over a fair amount of time.
- work in a wide range of environments not necessarily exhibiting regular structures, e.g. a building site may be mostly mud.

- be precise enough to assess design proposals based on the overlays produced.
- require minimal calibration of tracking devices.
- work both indoor and outdoor enabling maximal mobility, i.e. it should be possible to move around while tracking and to use the tracker at many different places.
- run in real-time and be able to handle abrupt camera motions.

Furthermore the tracking technology chosen must lead to a reasonably priced SitePack which in addition must be easy to carry and operate.

No existing tracking technology complies with all of these requirements. GPS technology has been used by numerous AR systems, but it is not possible to obtain the required precision unless combined with RTK GPS which restricts the mobility[7]. Using marker based approaches[10] on site is unpractical because of occlusions, the number of markers needed, marker registration in the same coordinate system etc.

Our work extends and combines previous work on real-time vision based feature tracking[9][14][15] to meet most of and relax some of the requirements listed above. Furthermore, previous work on real-time feature tracking has not addressed the issue of how to accurately initialize the feature tracker on site in real world scenarios, e.g. properly aligning a 3D model with the real world.

2.2. Initial Registration

All vision based approaches based on feature tracking must address the important problem of initially aligning the coordinate system of the model with the real world, as only the relative change in pose is computed by the tracker.

Our vision based tracker currently supports two ways of aligning a 3D model with the real world. Both methods require that some sort of correspondence between the 3D model and the real world be available. For this reason they cannot be applied in totally unstructured environments such as e.g. a field with nothing but mud.

Once the user is at the location from where he wishes to view the virtual 3D model, he points the SitePack in the desired direction and presses a button to initialize the tracker. This causes the video in the background of the Topos application to freeze and the 3D model is shown on top of the background as depicted in figure 2. To align the 3D model with the background video, the user can then mark a number of point correspondences between the background video and the 3D model. If the points are coplanar at least four points are specified [10], otherwise we use a simple method that requires at least six points[6], although better methods exist. When the correspondences have been marked, the 3D model is shown correctly aligned on top of the video stream, as shown in figure 3. In order to move the camera whilst maintaining the virtual overlay, the user simply points the SitePack in the initial direction, starts the tracker and subsequently moves the SitePack. This registration method has been applied with great success in indoor scenarios. However, consider the courtyard building site in figure 4. Most often the 3D model to be superimposed on the video stream does not contain significant texture detail in order to make sufficient point correspondences between the model and the video. Furthermore, identifiable features such as corners are often hidden by mud, tools or other equipment thus making the use of point correspondences alone non-feasible. For this

reason we offer a second method for doing the initial registration, based on identifying directions between the 3D model and the background video.

To recover the orientation of the camera, the user selects two orthogonal directions on the 3D model, drawn with an overlaying wire frame representation during the course of this registration method (figure 2). The two identical directions must be indicated on the video background. This is done by specifying vanishing points [6]. To specify a vanishing point at least two lines must be drawn. If the lines specified are detected to intersect at infinity, which is the case for the vertical directions in figure 2, the direction is recovered by exploiting that it must be parallel to the image plane of the camera.

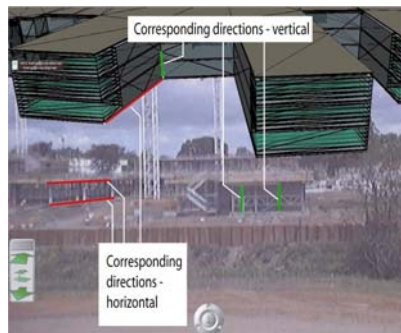


Figure 2: Shows recovering the orientation of a 3D model by specifying corresponding lines on the background image and the model.



Figure 3: Shows the result of the initial registration using corresponding lines for orientation recovery and manual recovery of the position.

Using these two direction correspondences, the orientation is easily computed. The position of the camera can be recovered by manually moving the camera using the manipulators supported by Topos. It could also be done by specifying two point correspondences between the 3D model and the background video. However, recovering the orientation manually is very cumbersome.

One could attempt to make this initial registration process automatic, however due to the limited, dynamic and often complex structures on a building site, this may not always be possible. For indoor scenarios, however, it is more likely to work.

Using the combination of automatic and manual techniques described above requires some skill of the user in addition to knowledge of both the construction site and the 3D model. However, once these are in place, the process is fast and easy.

The initialization processes described above requires the camera to be calibrated. If this is not done beforehand, a sensible approximation that works well in practice can easily be computed on site by using four rectangular points [4] or specifying the vanishing points of three orthogonal directions [6].

2.3. Homography Computation from Reference Images

The relative transformation of the camera between two images is computed from the homography[6] relating the two, and this homography is in turn computed from a number of feature correspondences. Our core real-time vision based tracker is similar to the methods in [6][9][14] that can be consulted for pseudo-code. Our method differs from previous work in the way reference images are captured and used to determine the transformation of the camera in each frame.

Each time a transformation is computed, an error is introduced. If the tracker computes the current camera pose based solely on the transformation in the previous image and the transformation between the current and previous image, this error will accumulate and thus cause undesirably drift[4][14]. Instead our tracker is based on relating the current image to a number of reference images, ie. computing the transformation between the current image and one of several reference images for which the transformation is known. The method described by Prince et al. [9] similarly relates the current image to a single reference image or reference mosaic, both of which must be captured in advance. Our method allows any number of reference images like the method of Stricker et al.[15]. However, contrary to the methods above, our reference images are registered automatically or on demand on site while tracking, thus avoiding the need to prepare the scene in advance whilst maintaining the extendible nature of the tracker.

The camera pose for each reference image is known as it is computed when the reference image is captured. Determining which reference image is to be used to compute the transformation in the current image is based simply on the maximal 2D overlap of each of the reference images with the current image. This is very fast, contrary to the reference image selection strategy of Stricker et al.[15]. The overlap is based on the homography computed in the previous frame and it is possible to do this 2D comparison as we are estimating a homography, i.e. a 2D projective transformation. The image used for the initial registration as described above is used as the first reference image. Additional reference images are needed if the overlaps between the current image and the existing reference images are too small which will cause the tracker to become unstable. Registration of a reference image can be done either manually by pressing a button (the camera pose for the reference image is then computed by the tracker) or automatically by a simple monitoring process that measures and ensures a minimal overlap between the current image and any reference image.

The camera pose computed for a reference image is contaminated by a small error. When subsequently relating the current image to such a reference image, the error will propagate to the pose estimated for the current image. However, since a small error is only introduced when registering a reference image and when relating the current image to the reference image, the error in the pose for the current image will be insignificant in most practical applications and significant drift will not occur.

Each time a camera transformation is estimated it is accompanied by an estimate of its quality. If this quality is below a certain threshold, the transformation is ignored and the one estimated in the previous frame is used instead. This has the practical consequence that if the tracking is lost due to e.g. abrupt camera motions, the camera can simply be pointed approximately in the direction in which the tracking was lost, and the model will jump into place.

3. Using the SitePack for Zone of Visual Impact Assessment

We have tried out the SitePack together with a group of landscape architects from the Scottish DLP, our user partner in the WorkSPACE project. During the last year they have been working on a site outside Edinburgh, covering an area of 78 acres.

One of the many challenges for the involved constructors and architects on a building site this size, is to communicate and visualize proposals to the contractors and the building owner. At the time of testing of the SitePack on site, the architects were working on a proposition plan for a courtyard garden, see figure 4. In the following month the proposition plan was to be carried out in a full scale mock up, so the building owner could get an impression of what the courtyard would be like when finished. Grass, hedges, trees etc. were to be planted full scale in the courtyard during a week, and then removed afterwards. This would be at the cost of a considerable amount of money, time and inconvenience for the construction workers.

The landscape architects were therefore interested in using the SitePack to visualize their 3D proposition plan of the garden on site. Primarily to facilitate evaluation of their own proposals before presentation, but also focusing on the possibility of using the SitePack in the future instead of full scale mock-ups.

During the test of the SitePack in the courtyard garden, a very direct result of the visualization was that the height of the hedges had to be reconsidered. From the viewpoint of the SitePack, chosen to be that of a standing person looking out of the window from the bottom of the courtyard, the superimposed 3D model showed that some of the hedges would actually block the view to the hills in the background.

Another issue of interest offered by overlaying the 3D model on the courtyard vista was the visualization of the visual impact of the finished three story building. When standing in the courtyard site there was an impression of semi-openness, due to the fact that only the first story of the building was partly finished. Watching the 3D overlay on the SitePack in the courtyard illustrated how the finished building would alter the courtyard impression into a feeling of enclosure and privacy.



Figure 4: Courtyard building site.

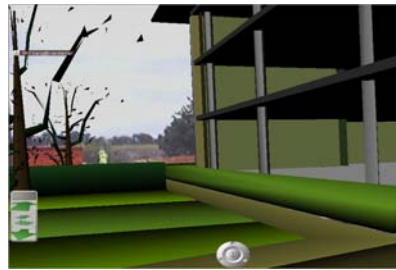


Figure 5: Shows the revised 3D model superimposed on the video in the courtyard garden.

In this scenario the initial registration was done on site by recovering the orientation from vanishing points and manually recovering the position. No preparation was required. Even though the illumination changed and construction workers moved around on the site, the tracking remained sufficiently robust throughout the session.

This would have been harder using previous techniques such as eg. a marker based approach described in section 3.1.

4. Experiences and Related Work

Although the test of the SitePack on site contributed to the evaluation of the landscape architects' design proposal, we also discovered some practical problems. Tablet PC hardware needs to become more powerful and the screen technology is not currently designed for outdoor use which means that screen visibility is reduced. However, compared to previous work such as MARS [7] and ArcheoGuide [15] the SitePack is much more mobile than the equipment required by those systems.

On site, the collaboration between the landscape architects was facilitated by the screen on the SitePack. They were able to directly discuss the impact of the 3D model proposition by sharing the same picture and focusing the camera towards views of interest and importance. Yet they could still sense the place by being there.

The vision based tracker proved to be sufficiently robust to occlusions and changes in illumination during a session. Contrary to other vision based trackers it does not require preparation of or intrusion on the site [9][10][15]. By combining the strengths of previous real-time feature tracking techniques [9][14] and using either an automatic or manual strategy for on-site registration of reference images we have obtained a feature tracker that is both extendible and does not suffer significantly from drift when used over time, contrary to previous work. Both of these features proved invaluable on site. If the camera was moved too fast or the video image contained too much motion blur, the tracker sometimes failed to estimate the transformation, however, as described previously, abrupt camera motions are handled in a gentle way.

The tracker only works for rotations or when tracking planar structures. This reduced the mobility of SitePack with respect to real-time overlays. However, most often the architects desired to remain in a certain position over a period of time to discuss a proposal anyway. Contrary to previous work on real-time feature tracking, we also described how to quickly and accurately register the camera to initialize the tracker at a new position directly on site in real world scenarios. This increased the mobility otherwise restricted by the homography-based vision tracker, although a tracker that allows full mobility while tracking by recovering 6 DOF is of course desirable. Prince et al. [9] describe how registration of the camera can be done by placing a marker in the first frame. However, in real life settings this is difficult to do precisely, as the position of the marker must be known in relation to the model and be placed exactly at this position in the real world. The main problems we experienced using the tracker was the high frequency jitter caused mainly by the noisy grained pictures captured by the web cam. This can be improved both algorithmically and by using a better camera. Another problem encountered on site was that if a pan of the SitePack was not close enough to a rotation, the tracker got confused by the parallax introduced in the movement of the tracked features. This was mainly a problem if the features being tracked were too close to the camera and can be solved by placing the web cam on a tripod, at the cost of some mobility.

5. Concluding Remarks

The paper has discussed an architect's SitePack appliance which features vision based feature tracking to support architectural work on site. We have described a successful example of use and the experiences gained. What distinguishes the SitePack software from previous systems is that tracking is extendible with insignificant drift and requires no preparation of the site – the site per se provides sufficient context for locating the user with overlay models precisely. Accurate initialization, which has not been addressed by previous work on real-time feature tracking, is achieved by a novel combination of automatic and manual techniques.

Acknowledgements

This work has been funded by the EU IST project WorkSPACE [www.daimi.au.dk/workspace/]. We would like to thank the entire WorkSPACE group for work on the Topos prototype and many fruitful discussions.

References

- [1] Azuma R.T. A Survey of Augmented Reality. In *Presence: Teleoperators and Virtual Environments* 6, 4 (August 1997), 355-385.
- [2] Billinghamurst, M., Kato, H. & Poupyrev, I. (2001). The MagicBook - Moving Seamlessly between Reality and Virtuality. *Computer Graphics and Applications*, 21(3), 2-4.
- [3] Billinghamurst, M., Kato, H., Kiyokawa, K., Belcher, D., Poupyrev, I. Experiments with Face to Face Collaborative AR Interfaces. *Virtual Reality Journal*, Vol 4, No. 2, 2002.
- [4] Gilles, S., Fitzgibbon, A.W., and Zisserman, A. Markerless Tracking using Planar Structures in the Scene. In *Proc. International Symposium on Augmented Reality*, Oct. 2000
- [5] Grønbæk, K., Vestergaard, P. P., & Ørbæk, P. (2002). Towards Geo-Spatial Hypermedia: Concepts and Prototype Implementation. In *Proc. of HyperText 2002*. Maryland: ACM.
- [6] Hartley R. & Zisserman A., *Multiple View Geometry*, Cambridge Univ. Press, 2000.
- [7] Höllerer, T. et al. Exploring MARS: Developing Indoor and Outdoor User Interfaces to a Mobile Augmented Reality System, *Computers & Graphics* 23(6), pp 779-785.
- [8] Höllerer, T., Feiner, S., and Pavlik, J. Situated Documentaries: Embedding Multimedia Presentations in the Real World. *Proc. ISWC '99 (Third Int. Symp. on Wearable Computers)*, San Francisco, CA, October 18-19, 1999, pp. 79-86
- [9] J.D.Prince S., Xu Ke, Cheok A.D, Augmented Reality Camera Tracking with Homographies, *IEEE Computer Graphics and Applications, special issue on Tracking*, November-December 2002
- [10] Kato H. and Billinghamurst M, Marker Tracking and HMD Calibration for a Video-based Augmented Reality Conferencing System, *Proceedings of the 2nd IEEE and ACM International Workshop on AR '99*.
- [11] Mackay, W. Augmented Reality: Linking real and virtual worlds. A new paradigm for interacting with computers. In: *Proceedings of AVI'98, ACM Conference on Advanced Visual Interfaces*, ACM Press, New York, 1998.

- [12] Milgram, P. & Kishino, F.A. "Taxonomy of Mixed Reality Visual Displays," Institute of Electronics, Information, and Communication Engineers Trans. Information and Systems (IECE special issue on networked reality), vol. E77-D, no. 12, 1994, pp.1321-1329.
- [13] Rekimoto, J. & Saitoh, M. Augmented Surfaces: A spatially Continuous Work Space for Hybrid Computing Environments. In: Proceedings of CHI'99 Conference on Human Factors in Computing Systems (Pittsburgh, Pennsylvania USA) ACM/SIGCHI, New York, NY, 1999, pp. 378-385
- [14] Simon G. and Berger M.-O., Pose Estimation for Planar Structures, *IEEE Computer Graphics and Applications, special issue on Tracking* , November-December 2002
- [15] Stricker, D. Tracking with Reference Images: A Real-Time and Markerless Tracking Solution for Out-Door Augmented Reality Applications, VAST 01