Real-time Planar World Modeling for Augmented Reality

Jonathan Ventura*
Department of Computer Science
University of California, Santa Barbara

Tobias Höllerer†
Department of Computer Science
University of California, Santa Barbara

Abstract

We posit that the challenge of complete visual modeling and tracking of the outdoor urban world can be made tractable by using the simplifying assumption of textured planar surfaces. While recent methods have demonstrated dense and complex reconstructions in some cases, we advocate simpler models which more efficiently and effectively capture the semantic structure of the scene. We argue that this structure is what will enable rich augmented reality interaction and annotation techniques. We describe three potential benefits of the planar modeling approach: 1) interactive mobile modeling and annotation; 2) fast and robust tracking and relocalization using oriented patches; and 3) scalable and incremental world model construction from ad-hoc user contributions. We consider several research questions and technical challenges which must be addressed to achieve mobile creation of piecewise-planar models.

Index Terms: I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Augmented Reality; I.4.8 [Image Processing and Computer Vision]: Scene Analysis—Tracking

1 Introduction

How can we map urban and indoor spaces for use in augmented reality systems in places where publicly available datasets such as satellite imagery and road-level captures are not applicable or available? There has been success in aggregating user-contributed photographs [19] but such data cannot be expected to offer sufficiently dense coverage everywhere. Instead, there is a need for single users to intentionally capture and contribute to a complete and useable world model. However, several questions need to be addressed before we can implement such a system. What model representation best enables interactive mobile modeling and tracking? Which representation is most useful for AR annotations and interaction?

Early researchers in automatic vision-based 3D reconstruction focused on recovering polyhedral models from line drawings [11]. Often the methodology worked up from line detection, to junction labeling, to a textured polyhedral model [9]. These early works modeled the world using only flat surfaces, and were based on reasoning about the visible edges of objects to produce a plausible geometric interpretation of the scene. Later, interactive systems such as Facade [5] offered tools for building polyhedral models of geometrically simple scenes by tracing their edges. These tools overcame the difficult problem of automatic scene interpretation by bringing a human into the loop to guide the reconstruction.

More recently, based on the success of robust keypoint matching [13] and the explosion of imagery on the Internet, research in vision-based modeling has shifted towards extraction of point clouds [19] and dense meshes [7] from multi-view image sets, even in real-time [17, 15]. These methods aim for fine detail and completeness in model creation.

What is lacking in these approaches is the underlying semantic structure of the scene, some of which was captured by the earlier polyhedral models. They work at a lower level than scene interpretation, by correlating and organizing image observations. General structures in the scene could be detected by post-processing the dense model, but this seems to require too much computation when what is needed for interaction is simply a flat surface. Also, as we discuss in Section 3.2, the wide baselines needed for accurate visual reconstruction on an outdoor scale may preclude an online structure-from-motion approach. Additionally, outdoor scenes tend to be highly dynamic, with lots of occlusion due to moving objects. Complete reconstruction of the scene might require a large of amount of redundant data capture, to aid outlier removal.

We envision a compromise approach which produces models more useful to augmented reality applications by extracting *structure* in the scene, as opposed to reconstructing its full geometric complexity. In this paper we argue for the approach of modeling a scene using textured planar surfaces, as has been used in several previous AR systems [16, 4]. Although simple in nature, planes are a good approximation to much of the complexity in a typical urban scene, as we discuss in Section 2. We consider the potential benefits of the planar assumption when creating (Section 3), tracking (Section 4) and indexing (Section 5) the model.

2 The Planar Assumption

If we assume that a surface is planar, we invariably will cause some errors in the reconstruction, since most objects are not truly flat. The question is, how much error will the planar assumption cause?

To analyze the error, we consider the reprojection error caused by erroneously placing a point on a plane, when it does not coincide with the plane. Figure 1 shows the two-dimensional case, where we assume all points lie on a line L. When viewed by a camera at c, the point p is placed at point q on the line L. This induces the reprojection error e when the point p is seen from c'.

We note that the reprojection error increases as p moves further away from the line along the normal. But, the reprojection error decreases as we move the point p and the line L further away from the camera along the camera's viewing direction.

The two-dimensional case easily generalizes to the usual three-dimensional case. This demonstrates that the planar assumption gets better when the surfaces are either more flat, or are further away from the camera. In the outdoor urban case, we look at building facades which typically are far away and roughly flat. Therefore, we do not expect planar models of buildings to cause significant errors in the overall computer vision system in most cases.

3 Interactive Mobile Planar Modeling

One benefit of planar model creation is that relatively little data is required to detect and store a planar surface, in comparison to dense multi-view reconstructions. In some cases, we can determine the vanishing point from lines in a single image [8]. With multiple images from a moving camera, feature and line matches are constrained by a homography which we can detect to extract the plane's parameters [2]. In this section we discuss our recent and ongoing work on plane detection in the mobile modeling case.

*e-mail: jventura@cs.ucsb.edu
†e-mail: holl@cs.ucsb.edu

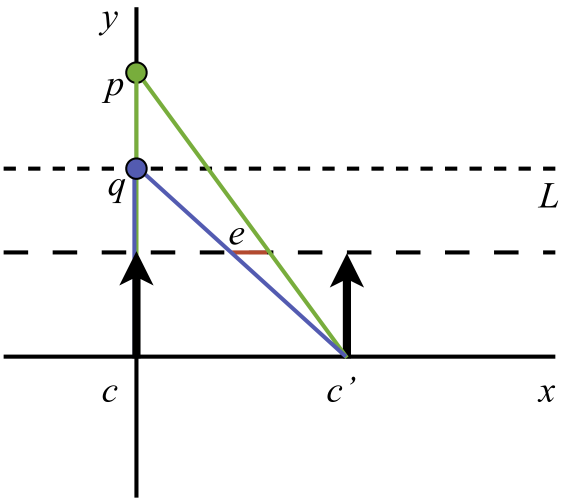

Figure 1: Diagram of reprojection error caused by the planar assumption, illustrated in the two-dimensional case. Point p is viewed by the camera with center c. By erroneously assuming that point p lies on the line L at point q, we induce the reprojection error e from the camera at c'.

3.1 Laser Rangefinder

One method of detecting planes and other structure in the environment is to use a depth-sensing instrument such as a laser rangefinder. We have experimented with a mobile system combining a laser rangefinder which operates at 10 Hz with a camera and HMD display. Planes can be delineated by sweeping the laser across them. Point readings are projected on to the ground plane, so that lines indicate vertical planes in 3D space. Wither's thesis work [25] describes robust methods for extracting these lines with RANSAC. Figure 2 shows the planes detected in one full sweep of the laser rangefinder in a courtyard. The extent of a plane in the camera image can be estimated by using a diffusion and segmentation process developed in joint work with Wither and Coffin [26].

The advantage of the laser rangefinder is that we can obtain accurate depth samples with little effort, using a compact mobile device. In future work we would like to develop methods for general plane detection using the laser rangefinder, without limiting ourselves to vertical walls.

3.2 Camera SLAM

A second approach we have tested is to automatically detect planar structure within a point cloud reconstructed using standard techniques. We tested this technique by using the output of a real-time SLAM system which tracks points in a video feed and generates a 3D point cloud [12]. By applying RANSAC we generate planar hypotheses and extract planes from the point cloud. Furthermore, the image of the plane in the camera frames can be identified using multi-view stereo matching [22].

One weakness of the structure-from-motion approach is that the accuracy and range of the system is highly dependent on the baseline between images. In the outdoor setting, the camera needs to move a significant distance before we can reliably estimate the depth of points. This makes for a difficult problem for feature tracking and matching. We have had success when running the SLAM system and detecting planes indoors at close range. However, it is difficult when dealing with far-away buildings outside to move with enough translation to reliably initialize point depth and plane estimates.

Also, this sort of specialized movement detracts from the quality of interaction in a mobile AR system. Ideally the system should be able to model the environment while the device is being used for other tasks which are more directly beneficial, e.g. when taking pictures of landmarks, or placing annotations.

3.3 User Assistance

Inspired by the aforementioned issues with structure from motion, we propose an alternate method for planar target identification, without any extra hardware, in which the user holds the camera directly facing the plane (e.g. a wall) to be estimated. In this case, all points on the plane can be assumed to have constant depth. With a single image, the scale of the scene is unknown. However, by assigning an arbitrary scale, we have sufficient information to initialize a local tracking map. This technique can be used to quickly make planar targets out of a textured wall, or any flat marker with natural features, such as a poster or a magazine.

One advantage of this method is that we reduce the burden of automatically detecting planes, by asking the user to directly identify them. The interaction is simple and only involves pointing the camera straight at a flat surface. This type of interaction is easy to understand and might be part of normal use of a cameraphone, e.g. when taking a picture of a building facade or a sign of interest.

In previous work in outdoor modeling, Piekarski and Thomas proposed several interactions using a mobile headset to identify and mark planes in a scene [16]. Although they mention the case of taking a frontal image of a wall, they focused on setting "working planes," by sighting down a wall from the side. They demonstrate how a complete geometric model of a simple building can be constructed *in situ* by intersecting several working planes.

For our scenario of visual tracking, it is more useful to capture images of walls taken from the front rather than the side, so that we capture the texture on the wall. However, the accuracy may suffer in the frontal case. One important research question is how accurately a user can line up a camera so that the plane normal coincides with the principal viewing direction. We hypothesize that the process of orthographic plane image capture may be improved by several features: visual aids, such as a grid overlay; output of orientation sensor readings; or "lucky imaging" where an image is automatically taken when the camera is level [1].

This method of directly identifying planes might be useful in combination with previously demonstrated techniques for automatically identifying scene structure. For example, we could initialize a SLAM system by imaging a flat surface, use the points to track the camera, and begin expanding the map automatically as the camera moves.

Additionally, we could ask users to take pictures of planar surfaces, without assuming that the camera coincides with the plane normal. We can still track from this surface by estimating a homography. As more images are added from different viewpoints (e.g. by other users), the location of the plane in 3D space can be determined. Gathering directed input from the user allows us to focus computation time on the structurally important parts of the environment.

4 Fast Tracking and Re-localization with Planes

4.1 Real-time Tracking

The basis of most tracking methods is the ability to match features across images. This matching is typically performed by assuming that the area around the keypoint is locally linear, i.e. planar, and thus is subject only to affine distortion under small movement. Accurate knowledge of point's normal is not needed to estimate this distortion, and most systems, e.g. Klein's PTAM tracker [12], assume that all normals point towards the camera.

However, these systems overlook the potential benefits of accurate knowledge of the plane normal. Indeed, several recent researchers have developed methods for determining locally planar

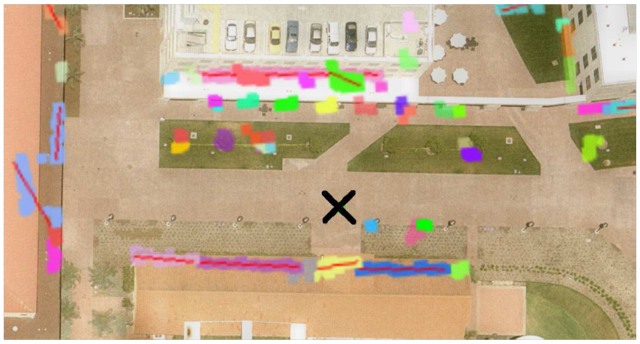

Figure 2: Planes detected using a laser rangefinder are overlaid on an aerial photograph. Each color represents one cluster of points corresponding to a plane. The cross marks the point of capture. Figure from Wither *et al.* [26].

keypoints and their normals [14], as well as approaches based on machine learning to automatically hypothesize surface normals [10]. Knowing that points lie on a plane gives greater confidence that they can be tracked using affine warps. From the plane orientation we can calculate the sampling rate of the surface's texture as seen from the camera. This could allow us to combine observations of the surface and eliminate redundant information. We also could use the point normal to better predict the effect of lighting and shadows on the feature's appearance.

4.2 Tracking Initialization and Recovery

By modeling an urban scene as a collection of planar textured surfaces, the camera re-localization problem becomes more similar to a large-scale image retrieval task. In the mobile SLAM case, the image retrieval index will likely be created and stored on the mobile device, so it must be space-efficient, and the image needs to be fast enough for real-time performance. In this section we explore keypoint-matching techniques for mobile image retrieval, which provide robustness to change in pose, illumination, and visibility.

There are two main approaches to keypoint description and matching. The first is computation of a complex, transformation-invariant descriptor from a single observation [13]. This approach requires signification computation time per keypoint, but offers a compact descriptor which can be matched despite a wide range of deformations of the image.

The second approach is to use a simple descriptor which can be matched over a smaller range of deformations, but is quickly computed [3]. To provide more robustness, usually thousands of warped observations of the keypoint are generated, and descriptors from all of them are combined for the matching step. This increases the memory requirements of the re-localization method.

A weakness of the above approaches is that they apply a pre-set filter to the image which is not adapted to the specific set of keypoints to be matched. For example, the Histogrammed Intensity Patch computes a probability distribution for each pixel in an 8x8 patch, and condenses each distribution into a 5-bit descriptor [21]. However, each pixel is treated equally – there is not an analysis to decide which bits are more important than others to the discrimination power of the descriptor. With such an analysis, we can further reduce the size of the descriptor by choosing the best binary decision functions.

Recent research on vector compression has found significant success by embedding vectors into a binary embedding such that the Hamming distance mimics with Euclidean distance [6]. The advantage of the binary embedding is low dimensionality and fast matching. In ongoing work, we have employed a method called spectral hashing [24] which analyzes the descriptor database to determine a set of binary tests by which we can produce a compressed bit-vector representation. In our experiments we have found that we can dramatically reduce descriptor size, down to 16 bits, while maintaining sufficient image matching performance.

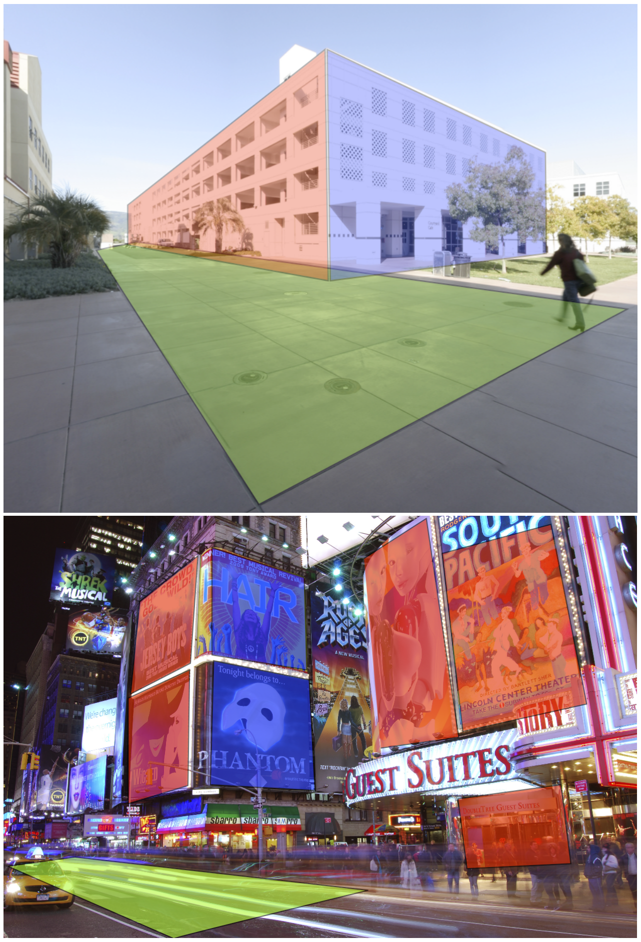

Figure 3: Two outdoor scenes roughly modeled using planes. These images were created by hand to illustrate how a scene can be represented in a piecewise planar manner.

5 Scalable Ad-Hoc Model Construction

A final, crucial component of urban modeling for augmented reality applications is the aggregation of multiple captures into a complete and usable model. In this section, we identify some challenges for combining planar surfaces observed by multiple users into a single piecewise planar model. We also consider how the model can be gradually refined and improved as more data is added.

Once multiple planes have been identified, mutual observances could be used to estimate the relative pose of planes. To recognize these mutual observances, we would need to be able to detect previously seen planes while tracking at same time (see Section 4.2). Recent developments in planar target tracking on mobile phones have used load balancing [23] and fast feature matching [20] to solve this problem.

Secondly, we would need a merging a step to find the transformation between the two planes and join them in to a single space. This could be performed with an initial linear pose estimation followed by non-linear error minimization. However, an important aspect of the merging process is finding a uniform scale, and avoiding drift [18].

We note that the planar structure provides a base for efficiently computing dense reconstructions using stereo methods. In the Facade system, Debevec *et al.* developed an algorithm for improved stereo by adjusting disparities once the rough planar estimate is known [5]. More recently, Newcombe *et al.* showed how, using a rough initial estimate provided by sparse structure-from-motion and meshing, fast GPU-based optical flow can be used to quickly generate an accurate dense reconstruction [15].

6 Overall System Proposal

Given our evaluation of the advantages and disadvantages of planar world modeling for augmented reality, we envision a system for outdoor modeling and tracking an urban setting which has the following components:

- Visual modeling of the environment as a set of a planar tracking targets
- Extension and updating of the model with a single camera image (possibly with location sensor data attached)
- Real-time camera tracking of targets with known orientation and texture
- Camera localization by image retrieval

7 Conclusions

In this paper, we have argued for an approach to environment modeling which estimates the idealized structure of a scene, as opposed to a precise geometric reconstruction. We have outlined several technical hurdles to overcome before we can achieve a highly scalable mobile AR system based on planar surfaces. Without introducing much error, such a model has several advantages in terms of implementation of the overall system, and support of AR tasks such as interaction and annotation.

Furthermore, we believe that the modeling techniques should be integrated into AR applications, such as building annotation, so that we can gain greater coverage and completeness through "crowdsourcing." By gathering observations of the environment in an ad-hoc but semi-guided manner, we can extract scene information from the areas which are of most interest to users of the system, and which may not be covered by other sources such as satellite imagery or street-level captures. Planar surfaces are a good starting point for feasible capture techniques to support this ad-hoc and user-driven approach to outdoor urban modeling.

Acknowledgements

This work was supported by NSF CAREER grant IIS-0747520.

References

[1] A. Adams, E.-V. Talvala, S. H. Park, D. E. Jacobs, N. Gelfand, J. Dolson, D. Vaquero, J. Baek, H. P. A. Lensch, W. Matusik, K. Pulli, and M. Horowitz. The frankencamera: An experimental platform for computation photography. In *SIGGRAPH '10: ACM SIGGRAPH 2010 Papers*, 2010.

[2] C. Baillard, C. Baillard, C. Schmid, A. Zisserman, A. Fitzgibbon, and O. O. England. Automatic line matching and 3d reconstruction of buildings from multiple views. *ISPRS Conference on Automatic Extraction of GIS Objects From Digital Imagery*, pages 69–80, 1999.

[3] M. Brown, R. Szeliski, and S. Winder. Multi-image matching using multi-scale oriented patches. In *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1*, pages 510–517, Washington, DC, USA, 2005. IEEE Computer Society.

[4] D. Chekhlov, A. P. Gee, A. Calway, and W. Mayol-Cuevas. Ninja on a plane: Automatic discovery of physical planes for augmented reality using visual slam. In *ISMAR '07: Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*, pages 1–4, Washington, DC, USA, 2007. IEEE Computer Society.

[5] P. E. Debevec, C. J. Taylor, and J. Malik. Modeling and rendering architecture from photographs: a hybrid geometry- and image-based approach. In *SIGGRAPH '96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 11–20, New York, NY, USA, 1996. ACM.

[6] A. Gionis, P. Indyk, and R. Motwani. Similarity search in high dimensions via hashing. In *VLDB '99: Proceedings of the 25th International Conference on Very Large Data Bases*, pages 518–529, San Francisco, CA, USA, 1999. Morgan Kaufmann Publishers Inc.

[7] M. Goesele, N. Snavely, B. Curless, H. Hoppe, and S. Seitz. Multi-view stereo for community photo collections. pages 1 –8, oct. 2007.

[8] R. M. Haralick. Using perspective transformations in scene analysis. *Computer Graphics and Image Processing*, 13(3):191 – 221, 1980.

[9] M. Herman and T. Kanade. Incremental reconstruction of 3d scenes from multiple, complex images. *Artificial Intelligence*, 30(3):289 – 341, 1986.

[10] S. Hinterstoisser, O. Kutter, N. Navab, P. Fua, and V. Lepetit. Real-time learning of Accurate Patch Rectification. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, 2009.

[11] T. Kanade. A theory of origami world. *Artificial Intelligence*, 13(3):279 – 311, 1980.

[12] G. Klein and D. Murray. Parallel tracking and mapping for small ar workspaces. In *ISMAR '07: Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*, pages 1–10, Washington, DC, USA, 2007. IEEE Computer Society.

[13] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110, 2004.

[14] N. Molton, A. J. Davison, and I. Reid. Locally planar patch features for real-time structure from motion. In *British Machine Vision Conference (BMVC)*, 2004.

[15] R. Newcombe and A. J. Davison. Live dense reconstruction with a single moving camera. In *CVPR*, 2010.

[16] W. Piekarski and B. H. Thomas. Tinmith-metro: New outdoor techniques for creating city models with an augmented reality wearable computer. *Wearable Computers, IEEE International Symposium*, 0:31, 2001.

[17] M. Pollefeys, D. Nistér, J.-M. Frahm, A. Akbarzadeh, P. Mordohai, B. Clipp, C. Engels, D. Gallup, S.-J. Kim, P. Merrell, C. Salmi, S. Sinha, B. Talton, L. Wang, Q. Yang, H. Stewénius, R. Yang, G. Welch, and H. Towles. Detailed real-time urban 3d reconstruction from video. *International Journal of Computer Vision*, pages 143–167.

[18] J. Repko and M. Pollefeys. 3d models from extended uncalibrated video sequences: Addressing key-frame selection and projective drift. *3D Digital Imaging and Modeling, International Conference on*, 0:150–157, 2005.

[19] N. Snavely, S. M. Seitz, and R. Szeliski. Photo tourism: exploring photo collections in 3d. In *SIGGRAPH '06: ACM SIGGRAPH 2006 Papers*, pages 835–846, New York, NY, USA, 2006. ACM.

[20] S. Taylor and T. Drummond. Multiple target localisation at over 100 fps. In *British Machine Vision Conference*, September 2009.

[21] S. Taylor, E. Rosten, and T. Drummond. Robust feature matching in 2.3μs. In *IEEE CVPR Workshop on Feature Detectors and Descriptors: The State Of The Art and Beyond*, June 2009.

[22] J. Ventura and T. Höllerer. Online environment model estimation for augmented reality. Oct. 2009.

[23] D. Wagner, D. Schmalstieg, and H. Bischof. Multiple target detection and tracking with guaranteed framerates on mobile phones. In *ISMAR '09: Proceedings of the 2009 8th IEEE International Symposium on Mixed and Augmented Reality*, pages 57–64, Washington, DC, USA, 2009. IEEE Computer Society.

[24] Y. Weiss, A. Torralba, and R. Fergus. Spectral hashing. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Advances in Neural Information Processing Systems 21*, pages 1753–1760. 2009.

[25] J. Wither. *Annotation at a Distance in Augmented Reality*. PhD thesis, University of California, Santa Barbara, 2009.

[26] J. Wither, C. Coffin, J. Ventura, and T. Hollerer. Fast annotation and modeling with a single-point laser range finder. In *International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 65–68, Sept. 2008.