

# A Mobile 3D City Reconstruction System

Brian Clipp\*  
UNC Chapel Hill

Rahul Raguram†  
UNC Chapel Hill

Jan-Michael Frahm‡  
UNC Chapel Hill

Gregory Welch§  
UNC Chapel Hill

Marc Pollefeys¶  
ETH Zurich/UNC Chapel Hill

## ABSTRACT

A city’s architecture gives physical form to the needs and aspirations of its residents. This fact is reflected in the complexity, variation and beauty of many cities. Current techniques for creating the architecture of virtual cities range from hand crafting by artists to highly automated procedural methods. In this paper we describe an efficient flexible capture and reconstruction system for the automatic reconstruction of large scale urban scenes. The system can be both backpack and vehicle mounted allowing the capture of interior or less accessible areas as well as large outdoor scenes.

**Index Terms:** 1.4.8 [Image Processing and Computer Vision]: Scene Analysis—Tracking; 1.4.8 [Image Processing and Computer Vision]: Scene Analysis—Sensor Fusion

## 1 INTRODUCTION

In the last years there is an increasing interest in the automatic generation of 3D models of entire cities. For example GoogleEarth and Microsoft VirtualEarth have started providing 3D models for a few points of interest. These models are currently captured with laser range scanning and then manually edited to achieve a higher visual quality and to correct for erroneous measurements. Alternatively 3D models are crafted by artists. As the level of realism of the architecture increases so do the labor costs and processing time. We propose an efficient capture system to capture the 3D-geometry of existing cities through computer vision techniques, leveraging the architectural properties inherent in existing urban environments to add realism. The system is able to deliver 3D reconstructions of large urban scenes with near real time.

In a previous project we recorded video of a city from a vehicle mounted camera system which included a highly accurate and expensive (\$150K) unit that combines a global positioning system (GPS) unit with an inertial navigation system (INS). With this system we recorded and reconstructed the geometry of most of Chapel Hill, NC as well as other locations. However, the high cost of this system makes its deployment on a wider scale impractical.

This paper focuses on a lower cost (\$13K) system we developed to record omnidirectional video of urban scenes as well as GPS and inertial measurements of the system’s movements. The system is modular and man portable, able to record both from a backpack mounting for interior areas and from an automobile for exterior recording. We reduced cost by utilizing consumer off the shelf sensors which, while less expensive, have significantly lower accuracy that poses a challenge to generating accurate geometric reconstructions and require new algorithmic solutions.

We break the problem of reconstructing the geometry of a scene from video into two parts. In the first part we perform structure from motion (SfM), simultaneously estimating the camera’s motion

and a set of salient 3D point features in the scene. These salient features are richly textured regions of the scene that can be uniquely and automatically identified, such as the corners of windows or doors. The second part uses the known camera poses to perform dense depth estimation, where we estimate the distance from the camera center to the scene for each pixel in the video. Combining depth information with the camera poses and images from the video, we generate textured 3D models.

In the remainder of the paper we review related existing systems for the reconstruction of cities from images, introduce our 3D city reconstruction system in detail, and explain the SfM process using only video data. We then describe our initial algorithmic SfM solutions for combining video data with other sensor types and the related challenges. Afterwards we discuss our dense geometry estimation and show the results of dense estimation from video only.

## 2 BACKGROUND

Recently, many research groups have been working to reconstruct urban environments from video. There is a large body of work in reconstruction from aerial imagery [11, 15, 30]. In recent years ground based sensors have attracted a lot of interest. Some approaches rely on active sensors such as LIDAR to recover depth information which is then aligned to images to form textured models [5, 10, 14, 27]. While these systems can create highly accurate 3D models, they are relying on active sensors which are typically more expensive than passive sensors like cameras. In this paper we focus on the reconstruction from video only or video combined with other passive sensors such as GPS and inertia sensors.

Active sensor systems and aerial reconstruction methods typically measure the sensor position highly accurately during the acquisition with additional sensors. Our proposed camera based system does not deliver camera poses or only rough estimates of the camera position. Accordingly we need to use the video data itself to measure the camera motion. There are two main classes of approaches to estimating the camera motion from video only. The first estimates the scene structure and camera motion directly from the measurements in an incremental process. It relies on the substantial work in multiview geometry [3, 13, 20, 21]. The second class of approaches uses the extended Kalman filter (EKF) to estimate the camera pose as the filter’s state [2, 26].

We have developed a real time 3D reconstruction system for urban environments [23]. The system leverages the parallel processing power of modern graphics processors to achieve real time performance while simultaneously achieving high accuracy. Camera poses are calculated in the system using measurements from SfM and a highly accurate GPS-INS system fused with an EKF. The GPS-INS system used, while highly accurate, is too large for man portability and too costly for deployment on a wide scale. For these reasons we are working to extend this work using low-cost, lightweight, commodity sensors.

In the 4D Atlanta project [24], Schindler *et al.* reconstruct the geometry of Atlanta based on historical photographs. Cornelis *et al.* [9] have created a city modeling system where the buildings are modeled as ruled surfaces perpendicular to the ground plane. The ground is a ruled surface defined based on the known geometry between the systems’s cameras and the wheels of the collection vehicle.

\*e-mail: bclipp@cs.unc.edu

†email: rraguram@cs.unc.edu

‡email: jmf@cs.unc.edu

§email: welch@cs.unc.edu

¶e-mail: marc.pollefeys@inf.ethz.ch

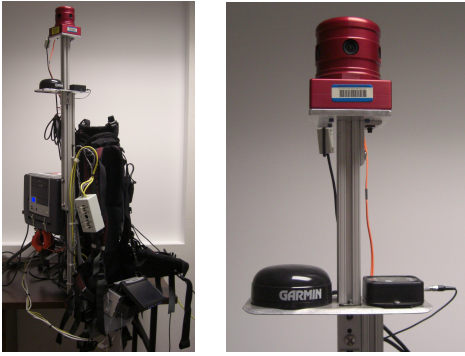


Figure 1: Backpack recording system (left). Rigid sensor head (right).

### 3 CAPTURE SYSTEM DESIGN

The recording system is designed to support three sensors, a Point Grey Research Ladybug omnidirectional camera, a Garmin consumer-grade GPS receiver with Wide Area Augmentation System (WAAS) capability, and an inertial sensor—a 3DM-G by Microstrain. The Ladybug is a multi-camera system consisting of six cameras of which five form a ring with optical axes roughly parallel to the ground plane when mounted and the sixth camera's optical axis pointing upwards. Each individual camera in the ring has a wide field of view covering roughly 73 degrees horizontally and 89 degrees vertically and a resolution of  $1024 \times 768$  pixel with a capture rate of 30Hz. When combined with the upward pointing camera, these provide video coverage of most of the sphere about the camera unit, except for the area directly below the camera. The GPS unit is accurate to approximately five meters under optimal conditions meaning many visible satellites, no or low multipath error etc. In practice this is highly unusual in urban environments and errors on the scale of 10 or more meters are more typical due to the urban canyon effect where very little of the sky is visible to the GPS receiver because of surrounding buildings. Finally the 3DM-G provides an absolute orientation measurement accurate to  $\pm 5$  degrees. The complete backpack recording system is shown in Figure 1.

We use a small form factor desktop computer with striped RAID drives to handle the video data rate and the amount of data captured over time. The system is capable of recording 15 frames per second uncompressed generating approximately 240 GB per hour of video. The sensors and computer are powered by a DC power supply and NiMH battery pack which support at least two hours of recording time. This is sufficient to capture several city blocks by simply walking around. When vehicle-mounted the system can be powered from an inverter and AC supply making the limiting factor the available storage space of 1.5 TB in the recording system's RAID drives.

The system can simply be converted from backpack to vehicle mount by disconnecting the sensor pod and connecting it to a car top mounting. Connecting the sensors together as a rigid system allows them to be extrinsically calibrated to each other yet convertible from backpack to vehicle mounted modes of operation. The collection system also doubles as a compute platform. Many of the algorithms we have developed rely on the parallel computing power of modern graphics processing units (GPUs). A relatively inexpensive desktop computer with a high-end graphics card provides the required computing power for a near real time reconstruction speed. In the future we aim for online reconstruction of urban environments using commodity sensors. However, currently recording the video data and other sensors data to disk takes most of the computer's bus bandwidth leaving little remaining for online reconstruction processing.

## 4 3D RECONSTRUCTION FROM VIDEO, GPS AND INERTIA DATA

In the previous section we described our mobile capture in detail which captures the data for the processing part of our system. The processing system performs two steps. First it estimates the camera using the video data or the video data jointly with the GPS and inertia data. Next the proposed system uses the camera poses and the video data to estimate the dense 3D scene geometry as a textured mesh. We will now discuss in more detail the two approaches to camera pose estimation (SfM), one which uses only video data as input and the other which is based on the extended Kalman filter (EKF) and fuses video data with GPS and inertial data.

The first step in both SfM methods is to extract the corresponding positions of salient features in multiple consecutive images. Salient features are typically corners in the image, dots or other geometric structures which have a high image gradient in two orthogonal directions in the image. These features can be uniquely identified and their corresponding positions tracked from frame to frame. We use the KLT tracking [18] because it generates accurate and stable correspondences over many frames of video. The system uses a fast implementation of the KLT-tracker on the graphics card [25]. The KLT-tracker delivers 2D tracks for the video.

### 4.1 Vision Only Reconstruction

Our purely video based monocular video system for 3D scene reconstruction takes the 2D feature tracks as its input. Using two images of the moving camera with five correspondences we estimate the relative camera motion using the five point method of Nistér [20] together with RANSAC [12]. The resulting camera poses and 2D correspondences are used to triangulate an initial set of 3D features. In addition to the triangulated 3D feature position we calculate the covariance of each feature which measures the certainty with which we know the feature position. Please note that each feature's covariance is independent of all other features it only depends on the certainty of the camera poses and the certainty of the features position in the images.

For a proper initialization it is critical that the camera moved sufficiently to observe depth dependent motion of the salient features tracked. The problem of the reasoning about the sufficient camera motion has been addressed by the use of uncertainty information [16, 17, 4]. Based on the results in [4], we employ a test that examines the shape of the confidence ellipsoids of reconstructed 3D points in order to determine the quality of the scene geometry.

Consider two corresponding image points  $x'$  and  $x''$  with associated covariance matrices  $C'$  and  $C''$ . The uncertainty ellipsoid  $C_{xx}$  for the corresponding 3D scene point  $X$  is then obtained as shown in Figure 2. By measuring the 'roundness' of this ellipsoid (i.e, the ratio of its smallest and largest axes), it is possible to obtain a measure of the accuracy of the reconstruction. Given the covariance matrix  $C_{xx}$  and the homogeneous vector  $X = [X_0^T X_h^T]^T$ , the covariance matrix for the distribution of the corresponding Euclidean coordinates may be obtained as

$$C_e = J_e C_{xx} J_e^T \quad (1)$$

where the Jacobian  $J_e$  is given by

$$J_e = \frac{1}{x} [I_{3 \times 3} - \frac{X_0}{X_h}] \quad (2)$$

Using the singular value decomposition of  $C_e$ , a measure of the roundness of the confidence ellipsoid is then obtained as

$$R = \sqrt{\frac{\lambda_3}{\lambda_1}} \quad (3)$$

where  $\lambda_1$  and  $\lambda_3$  denote the largest and smallest singular values respectively. By setting a threshold value for the roundness

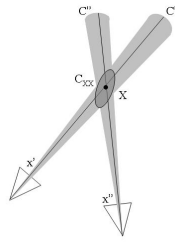


Figure 2: Initialization of the confidence ellipsoid of a 3D point.

( $T \approx \sqrt{1/10}$ ), it is then possible to eliminate points for which the uncertainty ellipsoid is stretched in the depth direction. This leads to a robust system initialization. After this initialization the system knows the position of the first two cameras up to a similarity transformation.

Afterwards each new frame is processed by first calculating the camera pose using RANSAC and the *three-point camera pose* estimation method [22]. This method uses three known, non-collinear, 3D features and their 2D image projections to estimate the pose of the camera. Our system uses the 2D feature tracks and the known 3D points from the initialization to estimate the new camera pose. Once the new camera pose is calculated additional 3D features are triangulated from new 2D tracks and their initial uncertainties are calculated. After initializing points, all previously existing 3D feature positions and uncertainties are updated using an independent EKF for each feature. This ensures that as additional 2D measurements of a 3D feature are received the SfM method incorporates this information into the 3D feature estimates in a recursive, efficient manner. Finally we can perform a bundle adjustment [1] on the total camera path and 3D feature estimates to improve their accuracy. An example of a textured 3D model generated from geometric, video only structure from motion can be seen in Figure 4.

## 4.2 Multi-Sensor Fusion

To perform SfM using vision and other sensors a model is required of how the measurements from each of the sensors contribute to the scene structure and camera motion estimate. Each of the sensors gives measurements in its own coordinate system, with differing units and accuracies. The camera returns images with units of pixels, the GPS returns position measurements in latitude and longitude and the inertial unit returns orientation as a quaternion representing orientation with respect to the earth.

We have chosen to model the set of sensors as a rigid body where measurements from the various sensors contribute to the estimate of the body's pose. An extended Kalman filter (EKF) estimates the pose of the body over time and the position of the salient 3D scene features. The EKF includes a state, state uncertainty model, process model and a measurement model. The state consists of the sensor system's position, velocity, orientation and rotation rate and the 3D position of each of the features being estimated. The state uncertainty model represents how accurately the filter has estimated its state and is encoded in a covariance matrix which includes both the expected accuracy of each state variable and also how those variables effect each other, e.g. the uncertainty in velocity is related to the uncertainty in position.

The process model describes the expected motion of the rigid sensor system and scene features between measurements (frames). Our process model uses a constant velocity, meaning that between measurements we assume that the body moves in a straight line in a rectangular coordinate system. We assume that scene features are static and consider moving features outliers to our model.

The measurement model generates expected measurements from the EKF state. For example, if a feature is at homogenous position

$X$ , the camera extrinsics are represented by the  $3 \times 4$  projection matrix  $P$  and the camera's intrinsics are represented by the  $3 \times 3$  matrix  $K$  then the expected homogenous feature measurement is its projection  $\hat{x} = KPX$ . GPS measurements are modeled simply combining the current sensor system position estimate and the translation from the center of the sensor system to the GPS unit. Similarly, orientation measurements from the inertia sensors are generated by combining the rotation of the sensor system and the rotation from the sensor system coordinate frame to the orientation sensor's coordinate frame.

The EKF breaks time into discrete intervals between measurements. Between measurements the uncertainty in the state grows due to the inherent inaccuracy in the process model. Based on the predicted state, the EKF generates a set of predicted measurements for each of the sensors as described above. In the correction phase of the EKF the predicted measurements are compared to the measurements from the sensors and the system state is updated based on their difference. The EKF weighs both the estimated accuracy of the state from the covariance matrix and the accuracy of the measurements to arrive at its update. A complete derivation and explanation of the EKF equations are beyond the scope of this paper. We suggest [6] for a more complete discussion on Kalman filtering.

A few problems exist in using the EKF to fuse vision based measurements with other sensors. The EKF assumes that the time each of the sensor measurements was taken is known. This doesn't seem like too much of an impediment until one considers that each sensor has its own clock with either its own estimate of local time or only a relative measure of time between its measurements. Each of these clocks also has varying accuracy. Each measurement may be assigned a time stamp based on the CPU clock when it arrives in a register on the CPU. However this ignores bus, operating system and other sources of delay. A typical solution for this would be the use of an external hardware trigger. These are not typically available in low cost commodity sensors and so measurement time alignment remains a challenge in our current work. One approach is to generate independent estimates of the system state over time from each of the sensors. Then correlation of the relative state change along the lines of [7] can be used to find an alignment between them in time.

Using only video and GPS for SfM has its own problems. Monocular vision-based SfM estimates cannot provide an absolute scale, orientation of world position because they are incremental in nature. To fuse vision with GPS using an EKF the vision measurements must be related to a single coordinate frame (typically that of the earth). This is done by initializing the EKF state with appropriate position, velocity, orientation and rotation rate parameters. Initial position and velocity are set easily enough by using the GPS measurements. Orientation and rotation rate are slightly more complicated since we do not have an absolute measure of either. We can estimate the sensor system's position utilizing any one of many monocular SfM algorithms and GPS measurements which are aligned in time. Given three position measurements from both vision and GPS when the sensor system is moving along a curve we can uniquely align the vision based poses and GPS measurements. This process is shown in Figure 3. We use the iterative closest point algorithm of [29] to align the two paths. From this we can derive an initial absolute orientation and rotation rate of the sensor system.

## 4.3 Dense Estimation

Structure from motion yields camera poses and a set of point features but we aim for a dense 3D model of the city's architecture. To generate the textured 3D model we use multi-view stereo matching between many images to give us depth information for each image in the video sequence. The depth information is the distance from the camera center to the scene for each pixel in an image. Combining this with the known camera poses gives the geometry of the

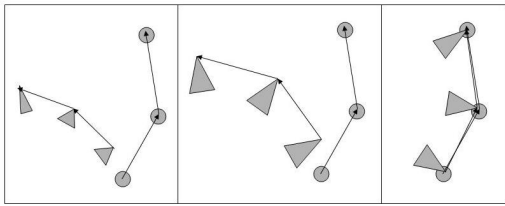


Figure 3: Alignment of vision based camera path and GPS. Left: camera path and measurements. Middle: camera path scaled to match GPS. Right: vision and GPS camera paths aligned using ICP.

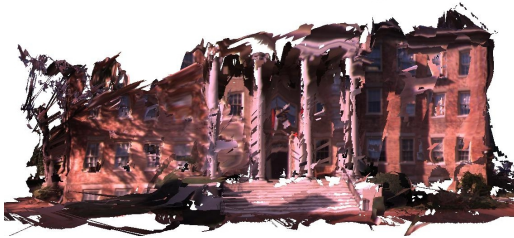


Figure 4: Video only 3D reconstruction with the back pack system

scene. We use a real time plane sweeping stereo algorithm which can be implemented efficiently on the GPU [8, 28].

Due to the small local computational effort in plane sweeping stereo it tends to generate noisy depth maps. Our system uses the redundancy of successive depth maps and combines their information to eliminate noise and achieve a higher accuracy. This fusion process [19] looks for consensus between the various depth maps about the location of a pixel in a chosen reference view. If the depth maps are in agreement as to the position of the feature imaged in the reference view's pixel then the depth is stored in the final fused depth map. However, if the depth maps are inconsistent then no depth is outputted into the final depthmap. Finally, a mesh is generated for each reference view using the fused depth maps and the texture from the reference view is applied to generate the final 3D model of the scene.

## 5 CONCLUSION

In this paper we have described our work on a mobile 3D city reconstruction system. We described the capture and reconstruction system and the related challenges for vision only SfM and fusion of video data with other sensors for 3D city reconstruction. Much work remains to be done in algorithms to fuse commodity GPS, video and inertial sensor data to automatically construct the architecture of an immersive virtual city environment.

## ACKNOWLEDGEMENTS

This work was supported in part by a grant from the David and Lucille Packard Foundation and the National Science Foundation and the DTO VACE program.

## REFERENCES

- [1] American Society of Photogrammetry. *Manual of Photogrammetry (5th edition)*. Asprs Pubns, 2004.
- [2] A. Azarbayejani and A. Pentland. Recursive estimation of motion, structure, and focal length. *IEEE PAMI*, 17(6):562–575, 1995.
- [3] P. Beardsley, A. Zisserman, and D. Murray. Sequential updating of projective and affine structure from motion. *IJCV*, 23(3):235–259, Jun-Jul 1997.
- [4] C. Beder and R. Steffen. Determining an initial image pair for fixing the scale of a 3d reconstruction from an image sequence. In *DAGM06*, pages 657–666, 2006.

- [5] P. Biber, S. Fleck, D. Staneker, M. Wand, and W. Strasser. First experiences with a mobile platform for flexible 3d model acquisition in indoor and outdoor environments – the waegele. In *ISPRS Working Group V/4: 3D-ARCH*, 2005.
- [6] R. G. Brown and P. Y. C. Hwang. *Introduction to Random Signals and Applied Kalman Filtering, 3rd Edition*. John Wiley and Sons, New York, 1997.
- [7] Y. Caspi and M. Irani. Aligning Non-overlapping Sequences. *International Journal of Computer Vision (IJCV)*, 48(1):39–52, 2002.
- [8] R. Collins. A space-sweep approach to true multi-image matching. In *CVPR*, pages 358–363, 1996.
- [9] N. Cornelis, K. Cornelis, and L. Van Gool. Fast compact city modeling for navigation pre-visualization. In *CVPR*, 2006.
- [10] S. El-Hakim, J.-A. Beraldin, M. Picard, and A. Vettore. Effective 3d modeling of heritage sites. In *4th International Conference of 3D Imaging and Modeling*, pages 302–309, 2003.
- [11] A. Fischer, T. Kolbe, F. Lang, A. Cremers, W. Förstner, L. Plümer, and V. Steinhage. Extracting buildings from aerial images using hierarchical aggregation in 2d and 3d. *CVIU*, 72(2):185–203, 1998.
- [12] M. Fischler and R. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Comm. of the ACM*, 24(6):381–395, 1981.
- [13] A. Fitzgibbon and A. Zisserman. Automatic camera recovery for closed or open image sequences. In *ECCV*, pages 311–326, 1998.
- [14] C. Früh and A. Zakhor. An automated method for large-scale, ground-based city model acquisition. *IJCV*, 60(1):5–24, 2004.
- [15] A. Gruen and X. Wang. Cc-modeler: A topology generator for 3-d city models. *ISPRS Journal of Photogrammetry & Remote Sensing*, 53(5):286–295, 1998.
- [16] S. Heuel. *Uncertain Projective Geometry: Statistical Reasoning for Polyhedral Object Reconstruction*. Springer, 2004.
- [17] K. Kanatani. Uncertainty modeling and model selection for geometric inference. *IEEE PAMI*, 26(10):1307–1319, 2004.
- [18] B. Lucas and T. Kanade. An Iterative Image Registration Technique with an Application to Stereo Vision. In *Int. Joint Conf. on Artificial Intelligence*, pages 674–679, 1981.
- [19] P. Merrell, A. Akbarzadeh, L. Wang, PhilipposMordohai, J.-M. Frahm, R. Yang, D. Nister, and M. Pollefeys. Real-time visibility-based fusion of depth maps. In *ICCV*, 2007.
- [20] D. Nistér. An efficient solution to the five-point relative pose problem. *IEEE PAMI*, 26(6):756–777, 2004.
- [21] D. Nistér, O. Naroditsky, and J. Bergen. Visual odometry for ground vehicle applications. *Journal of Field Robotics*, 23(1), 2006.
- [22] D. Nistér and H. Stewénius. A minimal solution to the generalized 3-point pose problem. *Journal of Mathematical Imaging and Vision*, 27(1):67–79, 2007.
- [23] 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. *IJCV special issue on "Modeling Large-Scale 3D Scenes"*, 2008.
- [24] G. Schindler, P. Krishnamurthy, and F. Dellaert. Line-based structure from motion for urban environments. In *3DPVT*, 2006.
- [25] S. Sinha, J.-M. Frahm, M. Pollefeys, and Y. Genc. Feature tracking and matching in video using programmable graphics hardware. *Machine Vision and Applications*, 2007.
- [26] S. Soatto, P. Perona, R. Frezza, and G. Picci. Recursive motion and structure estimation with complete error characterization. In *CVPR*, pages 428–433, 1993.
- [27] I. Stamos and P. Allen. Geometry and texture recovery of scenes of large scale. *CVIU*, 88(2):94–118, 2002.
- [28] R. Yang and M. Pollefeys. Multi-resolution real-time stereo on commodity graphics hardware. In *CVPR*, pages 211–217, 2003.
- [29] Z. Zhang. Iterative point matching for registration of free-form curves and surfaces. *IJCV*, 13(2):119–152, 1994.
- [30] Z. Zhu, A. Hanson, and E. Riseman. Generalized parallel-perspective stereo mosaics from airborne video. *IEEE PAMI*, 26(2):226–237, 2004.