Positioning, Tracking and Mapping for Outdoor Augmentation

Jayashree Karlekar Steven ZhiYing Zhou* Weiquan Lu Zhi Chang Loh Yuta Nakayama
Daniel Hii

Interactive Multimedia Lab, Dept. ECE,
National University of Singapore

ABSTRACT

This paper presents a novel approach for user positioning, robust tracking and online 3D mapping for outdoor augmented reality applications. As coarse user pose obtained from GPS and orientation sensors is not sufficient for augmented reality applications, sub-meter accurate user pose is then estimated by a one-step silhouette matching approach. Silhouette matching of the rendered 3D model and camera data is carried out with shape context descriptors as they are invariant to translation, scale and rotational errors, giving rise to a non-iterative registration approach. Once the user is correctly positioned, further tracking is carried out with camera data alone. Drifts associated with vision based approaches are minimized by combining different feature modalities. Robust visual tracking is maintained by fusing frame-to-frame and model-to-frame feature matches. Frame-to-frame tracking is accomplished with corner matching while edges are used for model-to-frame registration. Results from individual feature tracker are fused using a pose estimate obtained from an extended Kalman filter (EKF) and a weighted M-estimator. In scenarios where dense 3D models of the environment are not available, online 3D incremental mapping and tracking is proposed to track the user in unprepared environments. Incremental mapping prepares the 3D point cloud of the outdoor environment for tracking.

Keywords: Augmented reality, user positioning, robust tracking, shape matching, 3D mapping, sensor fusion.

Index Terms: H.5.1 [Information Systems]: Multimedia Information Systems—Augmented Reality; I.4.8 [Image Processing and Computer Vision]: Scene Analysis—Sensor Fusion, Tracking;

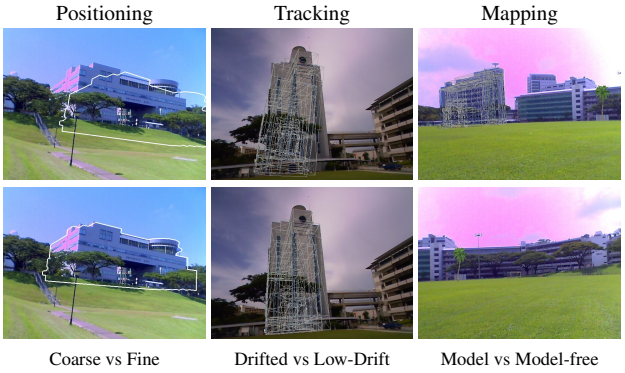


Figure 1: **First Column:** GPS and inertial sensors provide 6-DoF coarse estimate of the camera in outdoor environments (top). Non-iterative model-based shape matching approach is proposed to mitigate localization errors (bottom). **Second Column:** Feature based visual tracking causes accumulation of errors over time resulting in drifts of camera pose (top). Combined feature point and edge tracking reduces drift in pose estimation (bottom). **Third Column:** Dense modeling of the ever changing ourdoor environment is difficult and pose estimation should adapt to those dynamics. Extensible user tracking takes over as user moves from modelled environment (top) to non-modelled area (bottom). Virtual tree is augmented to illustrate the mapping and tracking.

1 INTRODUCTION

Outdoor augmented reality (OAR) combines the user's view of the real world with context specific information, such as text, images and 3D graphics.

The most important aspect of the OAR system is to identify the location and orientation of the user to retrieve the context so as to present him/her with context-aware information, thereby enhancing the user's awareness of the environment. These systems must run interactively and in real time to enhance the situational awareness. Accurate estimation of camera position and orientation in a global space is the most important aspect to provide such augmentation. Lack of accuracy can cause complete failure of coexistence of real and virtual worlds. In OAR systems, instantaneous 6-DoF user localization is generally achieved with position and orientation sensors such as GPS and gyroscopes. Initial pose obtained with these sensors is dominated by large positional errors due to coarse granularity of GPS data. Subsequent tracking is also erroneous as gyroscopes are prone to drifts and often need recalibration. In some applications accuracy of these devices may be adequate, whereas in others it is less than desired for true visual merging. Vision techniques are normally employed to mitigate sensor errors for accurate user localization and tracking in urban environments.

To overcome the practical limitations of different modalities in the context of outdoor environments, hybrid approaches are normally employed. These hybrid systems utilize data from position and orientation sensors as a rough estimate of the camera pose while vision technique refines it further. This paper presents one such novel hybrid model-based approach for outdoor mixed reality applications in which fine user positioning, robust tracking and mapping of the environment is achieved as illustrated in Fig. 1.

We propose a non-iterative model-based user positioning approach for OAR. The proposed one-step approach avoids costly rendering of the model at each iteration, hence minimizing initialization delay. Such automatic, accurate and fast initialization is necessary for practical OAR applications. The approach uses a contour/shape matching approach to obtain global camera pose. Coarse pose obtained from the system's sensors is refined by matching silhouettes of model data to the camera data (Fig. 1). Silhouettes obtained from the model and image data are parameterized and matched using shape context (SC) descriptors [3] as they are found to be the best non-parametric edge descriptors for 3D object matching [17]. These descriptors are invariant to translation, scale and rotational errors, which is very desirable for automatic alignment as coarse pose estimates are often dominated by large positional errors due to coarse granularity of GPS data. Final refined pose

*e-mail:elezzy@nus.edu.sg

IEEE International Symposium on Mixed and Augmented Reality 2010
Science and Technolgy Proceedings
13 -16 October, Seoul, Korea
978-1-4244-9346-3/10/$26.00 ©2010 IEEE

is obtained by constrained, reweighted Levenberg-Marquardt (LM) optimization (Section 3).

After localization, an extended Kalman filter (EKF) of constant velocity model is initialized and the system switches to tracking mode. In our system, further tracking is carried out with a pure vision based approach, as position/orientation sensors are employed during initialization and recovering from tracking failures only.

Once initialized, model-based tracking is carried out based on visual cues alone. Vision based sequential tracking approaches introduce drifts over time due to issues such as inaccuracies associated with camera calibration and feature tracking. The drifts are even more pronounced when the objects are poorly textured and occluded. This scenario is often exhibited in an urban context as buildings are homogeneous, poorly textured, often occluded by objects such as trees and vehicles. Keyframe based approaches offset the tracking drifts by periodically tracking with respect to reference frames which are obtained a priori or generated on the fly. Such keyframe based approaches are computationally demanding. In this paper we present a robust, keyframe-less sequential camera tracking approach for OAR. Our proposed approach achieves robust tracking by combining frame-to-frame and model-to-frame feature matches. Model-to-frame edge matching serves as a feedback mechanism to correct the drifts associated with camera tracking at each frame, thereby minimizing drifts from escalating over time. These different feature matches are appropriately weighted and fused using the M-estimator with the camera pose predicted from an EKF. A system of equations is then solved for camera pose by weighted least-squares (Section 4).

3D model generation of large outdoor environments is a research challenge. The complexity of the problem increases depending on the details that need to be modelled. Fully automatic, semi-automatic and manual modelling techniques are often employed depending on the required accuracy. The bigger challenge lies in maintaining these virtual models as real environments are dynamic and models also need to be dynamically updated to reflect the changes in the real environments. To adapt to these changes and do user tracking without prior knowledge, a model-free camera tracking approach is proposed in this paper. Such approaches are widely known in literature as *extensible tracking* [5, 7, 15, 16], in which the system extends its initial map by adding the new points, which are then later used for tracking in unprepared environments. In these approaches, the mapping of the environment is simultaneously carried out when model-based tracking is in progress. These mapping-while-tracking approaches can be broadly classified into two groups: keyframe based vs. incremental. Keyframe based approaches [15] employ a rigorous bundle adjustment (BA) to map the points robustly whereas incremental approaches do the mapping at each frame and are not robust. Incremental tracking approaches, also known as filtering or SLAM (simultaneous localization and mapping), achieve robustness by employing recursive refinement of 3D maps [5, 7]. An excellent analysis of both these approaches is presented in [25], where authors conclude that filter based SLAM frameworks are beneficial if small processing budget is available, however, BA optimization is superior elsewhere.

We employ a computationally light incremental mapping approach for resource poor OAR systems. Robustness is then introduced by fusing keypoints and edge tracking on visual data. We map the points incrementally at each frame as the user ventures into the non-modelled environment. Due to finite storage limitations on mobile devices, only features active in the current camera view are maintained and the rest are culled. Here we are looking at a particular scenario where the user explores the new regions without returning. The list of mapped features is refreshed from frame-to-frame (Section 5).

The results of user positioning, robust tracking and mapping are presented in Section 6. Limitations of the proposed system and further improvements are discussed in Section 7. The conclusion and future scope is outlined in Section 8.

2 Related Work

Many approaches ranging from sensor-based to pure vision based to hybrid ones have been proposed for outdoor augmented reality applications.

2.1 Initialization

Pure vision based techniques such as [2, 9, 13, 26] do image based object recognition to localize and track the users in outdoor environments. The user context in these approaches is acquired by querying the image database of labeled objects. On the other hand, hybrid approaches fuse data from different sensors such as camera, GPS and gyroscopes. In these hybrid approaches, the 3D georeferenced graphical models of the target serve as a context. Early work in model based automatic initialization is reported in [23]. The approach does model based automatic landmark detection and matching for rotational errors only. The approach presented in [21] does successive approximation of GPS data and edge tracking to converge to the correct pose, at the cost of initialization delay.

2.2 Tracking and Mapping

Assuming initializing, different hybrid approaches for tracking only are proposed in the literature. These hybrid tracking approaches mainly differ based on which natural features are used by the vision technique for tracking purposes. Approaches presented in [14, 29] use lines, [12, 20] are edge based while [1, 11, 24] use corners for tracking.

Inertial sensors suffer from drifts over long time and often need recalibration. Vision based sequential tracking approaches also suffer from drifts, normally arising due to inaccurate camera calibration, occlusion, incomplete or partial data, or limitation associated with different features itself. To achieve drift-free and robust tracking for longer sequences, keyframes or combined feature tracking is normally employed. Keyframe based tracking for unprepared environment is not feasible. A general solution for drift-free tracking is to combine multiple features such as corners and edges as proposed in [8, 4, 19, 22]. These approaches rectify errors associated with corner tracking by fusing edge matches, which serves as a feedback to minimize drifts.

To enable tracking in unprepared environments, many approaches for simultaneous tracking and mapping are proposed [5, 7, 15, 16, 24]. Approaches proposed in [5, 7, 15] are meant for small indoor AR workspaces to confine the mapping. In these approaches, initial 3D maps are further refined recursively or through bundle adjustments to achieve the robustness. An extensible tracking approach meant for mobile devices is proposed in [16], which is basically an adaptation of [15] for camera phones. However, these approaches are meant for small AR workspaces, and exploratory tasks such as OAR are not supported. The approach proposed in [24] is meant for outdoor AR and basically does the mapping to compensate for the drifts associated with inertial/magnetic sensors.

3 User Positioning with Shape Context Descriptor

In this section, we outline a novel model-based, non-iterative approach for automatic user positioning in urban environments. The coarse pose estimates obtained from the GPS and gyroscope are used to render the georeferenced graphical model. The silhouette of the rendered model is then extracted by thresholding. Without loss of generality, we assume that the texture-less, georeferenced models of the environment are available. Given resource scarcity on mobile devices, such models are more desirable as they take less disk space and are fast to render.

Extraction of the building silhouette from the camera image is more challenging. In indoor applications, object silhouettes are

generally extracted by "background" subtraction. Such a typical background is difficult to define and subtract in outdoor cases. The level-set based segmentation approach of [18] is computationally heavy and may still produce a distorted silhouette due to the occlusion of the target by trees, or the presence of ground plane or otherwise. To obtain meaningful data from the camera image, we segmented the image into sky and non-sky regions using a flood filling algorithm. The silhouette is then obtained by thresholding the mask, which contains the partial outline of the building and extra clutter. The silhouette/outline obtained from the image is the only available feature to match and refine the camera pose as other features such as edges tend to get cluttered due to large viewing distances. Reliable alignment is then achieved by matching the appearance of the model silhouette to the image outline. Fig. 2(b) illustrates the outline/silhouette extracted from the model and image for different initial poses.

With limited feature choices, model-based hybrid initialization with low level features is extremely challenging. Contour attributes such as gradient and curvature at different points can be used for possible matching purposes. Such attributes are of limited use in an urban environment where structures are predominantly cuboids in shape, which gives rise to straight lines in rendered images. Iterative closest point (ICP) [30] and distance transform (DT) based registrations also tend to get stuck in local minima and hence can be inappropriate for global localization.

Another alternative for contour matching is then to prototype the appearance of the contour obtained from the rendered model and match it in the camera data. One such prototyping of contours is proposed in [3] by the *shape context* (SC) descriptor. The descriptor characterizes a particular contour point with respect to all other contour points. These points are randomly sampled from the contour shape. Relative distances, angles and normalization make the shape context descriptor invariant to translation, rotation and scale respectively. Once model and image contours are parameterized with shape descriptors, one-to-one sample point correspondences are established (see Fig. 2(b)).

3.1 Silhouette Matching

Assume that the set, $P = \{p_1, \ldots, p_n\}$, $p_i \in R^2$, of n points represents the points on shape obtained from the model. Similarly, the set $Q = \{q_1, \ldots, q_m\}$, $q_i \in R^2$, of m samples represents the points on shape obtained from the image (Fig. 2(b)). Estimation of the alignment transform and registration is done by finding point matches using shape context as corresponding points on two similar shapes will have a similar shape context. For each point p_i from model contour the best possible matching point q_j from the image contour is obtained. The local cost of matching a point p_i to a point q_j is denoted as $C_{ij} = C(p_i, q_j)$. As shape contexts are distributions represented as histograms, the matching cost is obtained with χ^2 test statistic.

Forward matching (from model-to-image) of corresponding shapes is done by minimizing the shape context cost for each point p_i on model contour P to points q_j on image contour Q as,

$$C(P,Q) = \sum_{i=1}^{n} \min_{j} C(p_i, q_j). \tag{1}$$

Normally, the number of silhouette points n of model contour and m of image contour are different, giving rise to undesirable effects such as many-to-one matches or false matches. To obtain the unique correspondences, we employ the bidirectional tracking for matching silhouettes as,

$$C_{BT}(P,Q) = \frac{1}{2}[C(P,Q) + C(Q,P)]. \tag{2}$$

The results of bidirectional tracking are illustrated in Fig. 2(b) which gives unique matches. However, point matching based on shape context descriptor alone is not sufficient for reliable correspondences, especially in situations where extra clutter and/or partial shape data are encountered. Rigid registration based on these matches is infeasible as:

1. Shape context descriptors are defined with respect to all other contour points and partial information or extra information could change the distribution of points and hence the descriptors. This is normally the case for OAR as non-modeled elements such as pedestrians, vehicles, lamp-posts and trees could be present in camera data, giving rise to extra clutter as well as occluding the desired target.
2. Points which are close to each other on the model shape are often matched to points which are far away from each other on the image shape.
3. Iterative matching and registration can be employed to compensate these shortcomings but that means rendering and extracting the model shape many times which is costly in terms of computational power and delay in initialization.

Fig. 2(b) demonstrates the shape context based matching, where some model shape points are matched to the points from clutter.

The robustness of shape context based registration is then increased by introducing a figural continuity constraint proposed by [27]. The constraint states that, the two neighboring points on the model shape P should match to nearby points on the target shape Q. However, this constraint needs points on the model shape to be ordered. Once ordering of points p_i is done by using chain codes, neighboring matches are subjected to the figural continuity constraint. Point pairs not obeying the constraint are examined and correspondence having more matching error from the pair is detected as an outlier and excluded from pose calculations. The results of the figural continuity constraint are demonstrated in Fig. 2(b) in which the point matches not obeying the constraint are marked as red. From the remaining correct matches, camera pose is obtained by the constrained, weighted LM algorithm as described below.

3.2 Pose Estimation

To register the 3D model to the image, the world transformation $T \in SE(3)$ which is parameterized by a 6-dimensional vector $\xi = \{\theta_x, \theta_y, \theta_z, t_x, t_y, t_z\} \in R^6$ is obtained. The pose is estimated such that it minimizes the residual error r_i, that is

$$\xi = \arg\min_{\xi} \sum_{i} r_i^2. \tag{3}$$

Expressed in matrix form as

$$\xi = \arg\min_{\xi} \|\mathbf{f}(\xi) - \mathbf{b}\|^2 \tag{4}$$

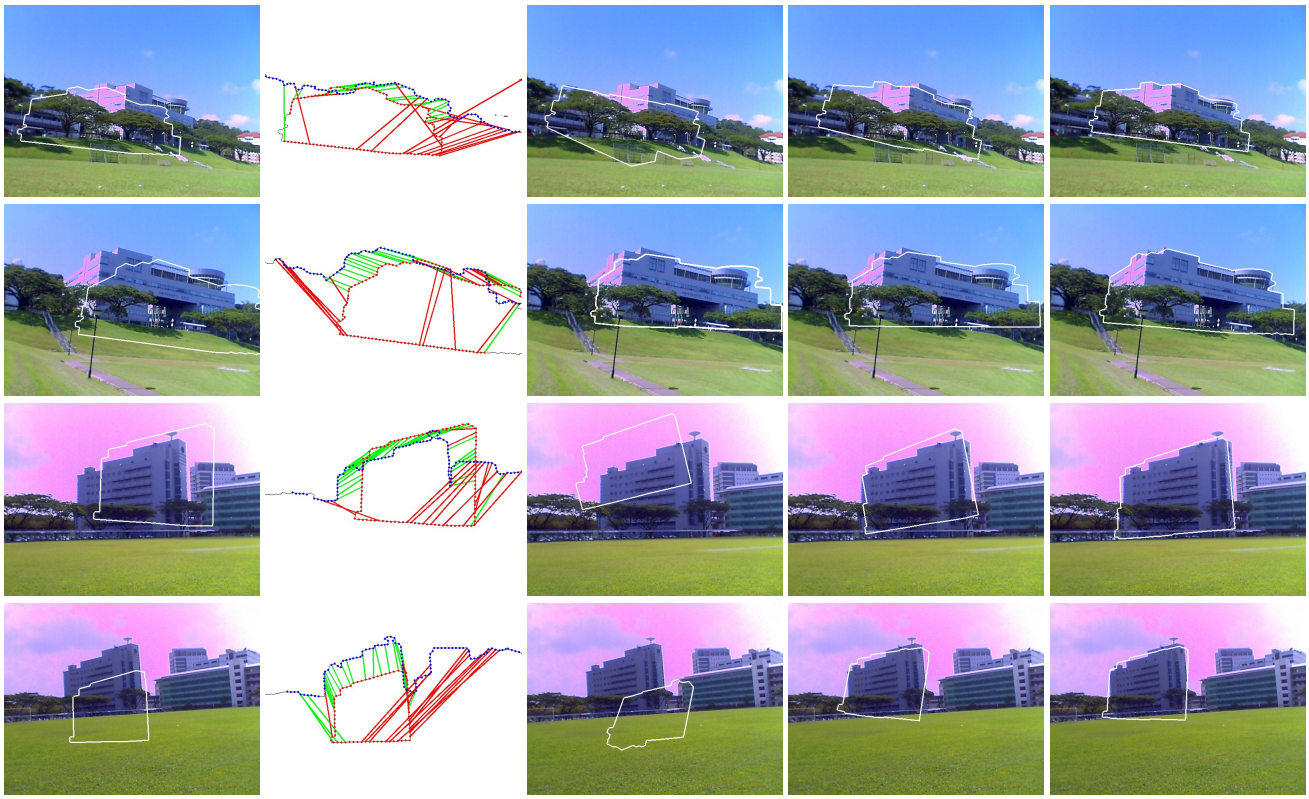
where $\mathbf{b}$ is a vector made of measurements obtained from Eq. 2 and $\mathbf{f}$ is a function that relates the camera pose to these measurements. The non-linear set of equations is then solved using an iterative Levenberg-Marquardt (LM) algorithm as:

$$\xi_{i+1} = \xi_i + \Delta_i$$

and step Δ_i is computed as:

$$\Delta_i = -(J^T J + \lambda I)^{-1} J^T \varepsilon_i \tag{5}$$

where J is the Jacobian matrix of $\mathbf{f}$ computed at ξ_i and $\varepsilon_i = \mathbf{f}(\xi_i) - \mathbf{b}$ denotes the residual at iteration i. Results of pose estimation using LM algorithm are illustrated in Fig. 2(c). The values obtained for ξ are plotted in Fig. 3. The estimated camera pose ξ has large values for angle and translation parameters. The cause of misalignment



(a) Coarse camera pose (b) SC matching (c) LM (d) CLM (e) WCLM

Figure 2: Illustration of coarse-to-fine user positioning: (a) Initial sensor estimates are used to render the georeferenced model; (b) Silhouettes are extracted from image (blue) and model (red) to do matching with shape context (SC) descriptor. Green point matches represent the correct matches while red ones are outliers; Final user pose is obtained with: (c) unconstrained LM algorithm; (d) constrained LM (CLM) algorithm; and (e) weighted, constrained LM (WCLM) algorithm.

was the presence of outliers which were still present even after imposing the figural continuity constraint. Such outliers can not be totally avoided due to presence of occlusion and extra data.

We can decrease the influence of outliers if they cannot be totally avoided by imposing an extra constraint on the estimated value of ξ. Normally, accurate user position is in vicinity of the coarse camera pose. Accuracy of GPS data in urban environment is typically within $5-10$ meters. Further parameterization of the GPS error with Guassian distribution has been reported in [21], where they found that the error has the standard deviation $\sigma = (1.9m, 4.3m)$ in east-west and north-south direction respectively. To be able to initialize with high likelihood, the exact user position could be anywhere in an area encapsulated by an ellipse of $3\sigma = (5.7m, 12.9m)$ around the reported GPS location. Assuming isotropic errors in both directions, the worst case user position could be $(\pm 15m, \pm 15m)$ from the coarse GPS location. Similarly, worst case errors for orientation data in pitch, yaw and roll are assumed to be $(\pm 15^o, \pm 15^o, \pm 15^o)$ in X, Y and Z direction respectively. With these bounds, the results of the constrained LM (CLM) optimization are given in Fig. 2(d) and 3. While the results are within bounds and very near to the desired camera pose, effects of outliers are still visible.

As observed, the least squares formulation is very sensitive to outliers. To subsidize the impact of outliers on the final pose estimation, we employ M-Estimators. Instead of minimizing Eqn. 3, one can minimize its robust version,

$$\sum_i \rho(r_i) \tag{6}$$

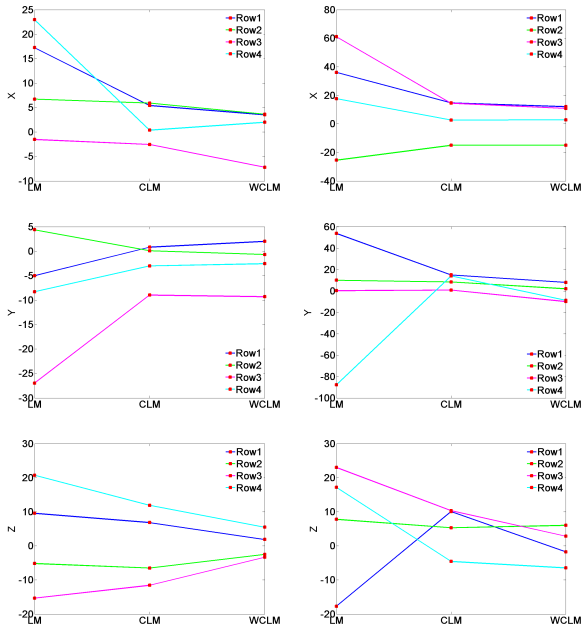
where ρ is an M-estimator that reduces the influence of outliers [31]. Effect of M-estimators is incorporated into minimization by simply weighting the residuals r_i with weight ω_i. The matrix W can be taken as $W = diag(\ldots \omega_i \ldots)$. In case of LM estimation scheme, modified Δ_i is computed as:

$$\Delta_i = -(J^T W J + \lambda I)^{-1} J^T W \varepsilon_i \tag{7}$$

We have used the Huber function [31] to estimate the weights ω_i. The results of re-weighted, constrained LM optimization are illustrated in Fig. 2(e). Results of Fig. 3 clearly demonstrate the robustness of the weighted-constrained LM (WCLM) algorithm producing the most optimal user pose.

4 Combined Corner and Edge Tracking

After correcting the initial camera pose, the system switches to tracking mode. The constant velocity EKF is initialized and updated at each frame which serves two purposes: one to provide an estimate of the camera pose for the next frame to fuse feature matches, and another to smooth out the jittering effect of tracking. The drifts associated with vision based tracking are well known, which arises due to inaccurate camera calibration, presence of occlusion, incomplete or partial data. To achieve drift-free and robust tracking for longer durations, two popular approaches are: keyframe based tracking and combined feature tracking. In keyframe based tracking, drifts over longer sequences are reset by tracking periodically with respect to keyframes. Such keyframe based approaches work well in prepared environments or for small



Angle plots in degrees Translation in meters

Figure 3: Estimated camera poses with different LM variants. Results corresponds to rows of Fig. 2.

workspaces. On the other hand, combined feature tracking approach is more general and equally applicable for controlled indoor environment as well as for unprepared outdoor tracking.

In combined feature tracking approach, multiple features such as corners and edges are fused to increase the robustness and tracking accuracy. Corner tracking is normally performed on frame-to-frame camera data. Similarly, edge tracking is performed on model-to-frame data. One can further use model-to-frame corner matching as reported in [8]. Utilizing such matching entails the use of 3D textured models which we have avoided here to keep them simple from a rendering point of view. Moreover, inaccurate texture stitching can lead to feature matching errors.

The frame-to-frame corner matches are denoted as C_{FF} and model-to-frame matches are denoted as C_{MF}. Keypoint matches of C_{FF} captures the major camera motion while edge tracking of C_{MF} provides feedback of misalignment to correct the drifts, if any. Such combined tracking has been successfully used for tracking articulated [8] and complex rigid objects [22, 28]. An approach proposed by [4] fuses corner and edge tracking in a unified manner. All these approaches are proposed for indoor environments and their application to outdoor environments is challenging.

The outdoor environment poses different challenges than controlled lab environments. Most of the time, meaningful edges are difficult to extract for feasible model-to-frame edge matching. To provide reasonable feedback to avoid drifts in tracking, the silhouette of the model is matched to the partial silhouette extracted from camera data as discussed in Section 3. The moving edges algorithm of [6] is used to obtain C_{MF} matches while C_{FF} matches are obtained by using Harris corner detector and matching [10]. Fig. 4 illustrates combined tracking process.

Region based corner matching suffers from occlusion and mismatches arising due to repetitive structures present in man made environments. Similarly, edge tracking suffers from the aperture problem. The moving edges algorithm captures motion only in the direction perpendicular to an edge while motion along the edge is not detected. To reduce the influence of different feature tracking algorithms, matches are appropriately weighted before fusing. Weighting is done using M-estimators. Camera pose for the current frame is predicted from the EKF and the Huber estimator [31] is used to generate weights. The final camera pose is predicted by weighted least-squares system of Eq. 4.

Figure 4: Combined feature tracking is in progress. Extracted silhouette from image is overlaid in white color. Edge correspondences obtained from model-to-frame tracking C_{MF} are illustrated with red color while green ones corresponds to frame-to-frame region based corner matching C_{FF}.

5 Incremental Mapping

The comprehensive 3D modelling of outdoor areas is a difficult task. However, 3D models of some target interest areas, such as heritage centers and tourist places, are achievable. Assuming that such 3D models of some popular targets are available, we propose extensible tracking for general outdoor areas of which prior knowledge in the form of images/models is not available. Tracking for unknown areas is accomplished from known targets with a simultaneous tracking and mapping approach. That is, normal model-based tracking will be done as usual while simultaneous mapping of the remaining part will be carried out from tracked information. That way, a map of the unknown environment is maintained. When the user explores more general areas apart from targeted ones, previously mapped data is used to track the camera. A parallel tracking and mapping approach presented in [15] achieved fast and accurate mapping based on keyframes while the approach of [7] does incremental tracking. Both the approaches were targeted for small, indoor AR spaces.

We adopt a computationally light and less resource demanding incremental mapping as opposed to the keyframe one for OAR. However, maps constructed with incremental tracking are not robust as camera tracking drifts over time. To provide robust tracking and mapping we fuse different features to track the camera, similar in spirit to that of the previous section. However, in the absence of any wireframe model of the environment, we resort to frame-to-frame silhouette matching as opposed to model-to-frame silhouette one. Frame-to-frame feature matches C_{FF} now consists of two features: one based on keypoints and other based on silhouette tracking. Keypoint matching is the same as that of the previous section while silhouette tracking is carried out with SC descriptor matching as opposed to edge tracking to take care of aperture issues. The mismatches arising from SC descriptor are avoided by imposing extremely stringent matching threshold and confining the search window to a small area around the contour point.

To keep the mapped data to a minimum, the interest points vis-

ible in current view of the camera only are maintained. The list of interest points is updated at every frame as new features are added while non-visible ones are removed from it. Hence, at a time not more than 1000 points are mapped and maintained.

Once camera pose from model points is obtained, the remaining keypoints belonging to the non-modelled area are mapped by minimizing forward projection errors. Projection of homogeneous scene point $\mathbf{X}$ in two views are $\mathbf{x}_1 = \mathbf{P}_1\mathbf{X}$, and $\mathbf{x}_2 = \mathbf{P}_2\mathbf{X}$, where projection matrix $\mathbf{P}_1$ and $\mathbf{P}_2$ are known. Forward projections of x_1 and x_2 are:

$$A\mathbf{X} = \begin{bmatrix} x_1 p_{13} - p_{11} \\ y_1 p_{13} - p_{12} \\ x_2 p_{23} - p_{21} \\ y_2 p_{23} - p_{22} \end{bmatrix} \mathbf{X} = 0 \qquad (8)$$

where p_{ij} is the j-th row of P_i. Non-zero solution for $\mathbf{X}$ is obtained by decomposing A into SVD and retaining the vector corresponding to smallest eigen value.

The overall algorithm for positioning, robust tracking and mapping is outlined below.

Algorithm:

Initialization:

1. Coarse camera pose ← GPS and gyroscope
2. Extract silhouettes from image and rendered model
3. Obtain correspondences between image and model silhouettes with bi-directional matching with shape context (SC) descriptors (Eqn. 2)
4. Obtain camera pose using constrained, re-weighted LM (Eqn. 7)
5. Initialize EKF

Robust Tracking:

1. Render the model with the camera pose obtained at $t-1$
2. Obtain frame-to-frame matches C_{FF} with keypoint/patch matching
3. Obtain model-to-frame matches C_{MF} with edge tracking
4. Predict the camera pose at t with EKF
5. Fuse different feature matches C_{FF} and C_{MF} by Huber function w.r.t. camera pose obtained from EKF
6. Estimate the pose at t with weighted least squares
7. Update EKF

Incremental Mapping:

1. Obtain camera pose with model-based tracking by frame-to-frame matching of keypoint and edge features
2. Map non-modelled feature points by forward projection
3. Cull the points not active in current camera view
4. Use mapped points to track the user in arbitrary environment

---

(a) (b) (c)

Figure 5: Results of one-step, model-based user positioning for different scenarios. The user is localized by shape context matching and WCLM optimization. (a) Coarse pose obtained from position and orientation sensors; (b) Corrected pose with superimposed silhouette; (d) Corrected pose with augmented wireframe of the model.

6 Experimental Results

6.1 Hardware

The hardware system specification used in our setup consists of the following devices. The Logitech QuickCam Pro 5000 camera, with outer casing removed, is used for capturing images at 320×240 resolution. The inertial sensor device used for orientation measurement is OS5000-S. The gyroscope is tightly coupled to the camera. The position sensor is the HOLUX M-1000 wireless Bluetooth GPS receiver having 1 second update rate. The system is targeted for the UMPC (Fujitsu U2010 1.66Hz, single core with INTEL GMA 500 graphics card) and runs at 9 frames per second. The performance varies depending on the polygon count of the model used for tracking. We have used two virtual buildings approximately consisting of 1700 and 2300 polygons respectively.

6.2 User Positioning

Results of automatic user positioning with shape context based silhouette matching under extreme variations are presented in Fig. 5. The silhouette from the image is extracted by a flood filling and thresholding algorithm while that of model is obtained with thresholding. 3D transformation parameterized by the 6-dimensional vec-

Figure 6: Visual representation of robust camera tracking results. Top row demonstrate the results of combined feature tracking while bottom row illustrates tracking based on keypoints alone. Video sequence consists of 1276 frames.

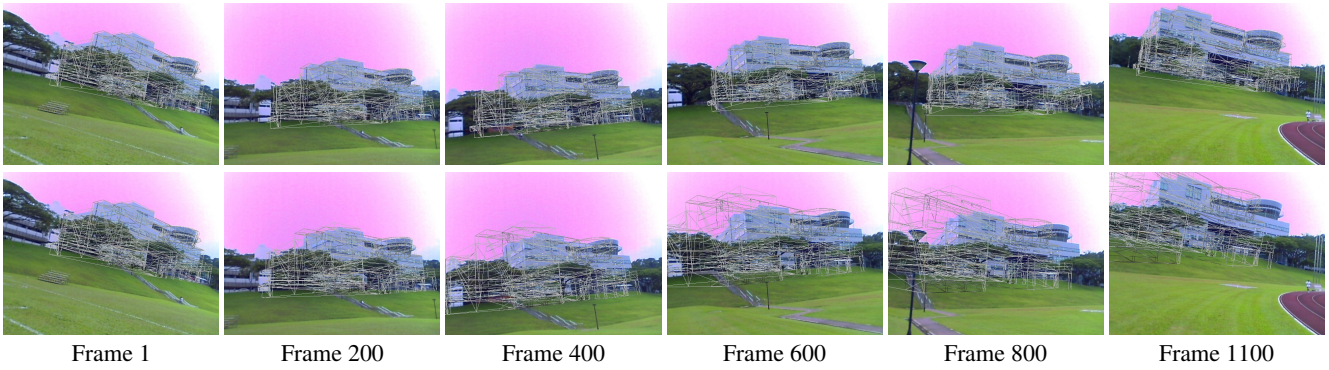


Figure 7: Visual representation of robust camera tracking results. Top row demonstrate the results of combined feature tracking while bottom row illustrates tracking based on keypoints alone. Video sequence consists of 1100 frames.

tor ξ is estimated using the iterative, re-weighted, constrained LM optimization.

Rough pose estimates obtained from the sensors are dominated by translation and scaling errors due to coarse granularity of the GPS data (Fig. 5(a)). Variations exhibited by these errors from time-to-time and day-to-day make it difficult to parameterize them. Simple parameterization based on Gaussian distribution was done in [21]. This parameterization provides upper and lower bounds on the probable position and makes the search of correct pose feasible. Alignment results demonstrate the invariance of shape context based matching and WCLM optimization towards those errors (Fig. 5(b) and (c)). The accuracy of correct camera pose estimation entirely depends on the quality of the matches between two shapes. Hence it is very essential to detect and reject outliers.

6.3 Tracking

Once initialized, the system switches to tracking mode. After initialization, data from the camera alone is used for tracking. Cameras with fish-eye lenses seem to provide better tracking results for longer video sequences. However, in such scenarios drifts are perceived later rather than sooner. Moreover, irrespective of the camera lens, drifts are inherently associated with visual tracking.

Major camera motion is captured by frame-to-frame keypoint matching C_{FF}. Drifts associated with C_{FF} matching are minimized by estimating misalignment at every frame. The misalignment is estimated by matching the silhouette of the model to that of the image. To keep the processing simple, edge-tracking is performed to do silhouette matching and is denoted as C_{MF}. Drifts are estimated at every frame and fused with keypoint matches. These matches basically provide the feedback to the tracking mechanism. C_{MF} matches suffer from the aperture problem associated with edge-tracking as motion cannot be perceived along the edges. Hence, C_{FF} and C_{MF} matches are appropriately weighted before final pose estimation. These weights are generated by predicting pose from EKF and M-estimator. We used Huber function estimator to fuse the matches.

Subjective results for keypoint based tracking and combined feature tracking are tabulated at different instances in Fig. 6 and 7. Video sequence of Fig. 6 has over 1200+ frames while that of Fig. 7 consist of around 1100 frames. The first row in both the figures presents the results of the proposed tracking algorithm while the last row shows results obtained by keypoint tracking only. The keypoint based tracking starts drifting early and never recovers for both the sequences while low-drift tracking is achieved with the combined feature tracking approach even after 1000 frames. Edge tracking prevents the drift from escalating over time. In C_{FF} based tracking, negligible drift is observed till 100 frames for both the sequences. The tracking results are provided in the supplementary video material.

Fig. 8 presents the quantitative results of the user's position extracted from the two methods for video sequence of Fig. 6. Feature tracking results are then compared with raw GPS data (red trajectory). The blue trajectory depicts the pose obtained with our proposed algorithm. The position data obtained from GPS sensor exhibits roughly 5 meters drift in the east-west direction while it is 10 meters in the north-south direction. The figure clearly demonstrates the drifts in position estimation with C_{FF} based tracking (green trajectory) right at the beginning and not following the coarse path obtained from GPS. The orientation results obtained with the two approaches are plotted in Fig. 9. Both the approaches produce com-

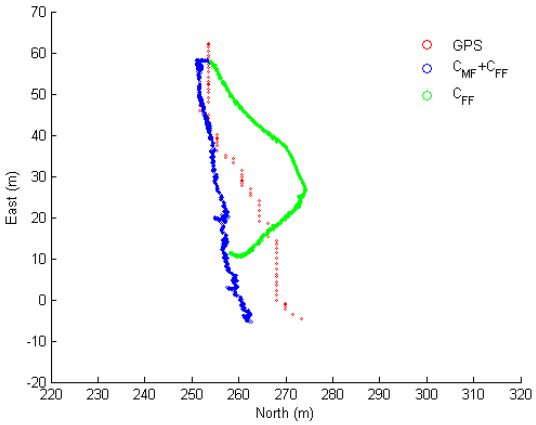


Figure 8: Camera tracking results projected in local coordinate plane as east and north map points in meters for the video sequence of Fig. 6. Raw GPS position (red), results of proposed $C_{MF}+C_{FF}$ combined feature tracking (blue) and results of C_{FF} tracking (green) alone.

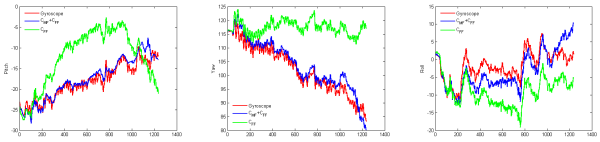


Figure 9: Results of camera orientation obtained with $C_{MF}+C_{FF}$ (blue) and C_{FF} (green) tracking for the video sequence of Fig. 6. Orientation data obtained from gyroscope is plotted in red for comparison.

parable results for first 100 frames. Then the results of C_{FF} based tracking starts deteriorating and digressing from the proposed combined tracking approach and gyroscope data.

We also observed that the proposed combined feature matching and tracking approach is pretty robust to initialization errors arising from user positioning approach of Section 3. This is expected, as edge tracking provides estimate of misalignments which is corrected as the tracking progresses.

6.4 Failure Recovery

In OAR, mobile devices are either head-mounted or handheld, which makes them susceptible to jerky motions, causing tracking failures. In such scenarios, re-initialization is needed to maintain the tracking. We illustrate the failure recovery results for SC based matching approach.

Under sudden motion, local feature matches obtained by the C_{FF} and C_{MF} tracking produce negligible feature correspondences, causing no updates in camera pose, hence leading to tracking failure. Tracking should be resumed and the model should lock back when camera recovers from such disturbances. Automatic, robust and fast re-initialization is must to resume the tracking. Fig. 10 and 11 illustrates the robustness of the SC based matching under sudden motions. Silhouettes of model and image are extracted and matched to re-initialize the camera pose. To ensure that the silhouette extracted from image is not noise, we also employ keypoint based matching C_{FF} between the frames when tracking stopped and the current camera frame for rapid and robust pose recovery.

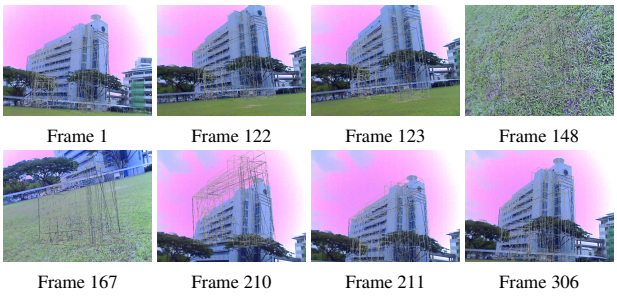


Figure 10: Results of failure recovery. Tracking starts at frame 1 and continues till frame 122. Sudden motion at frame 123 causes no updates in camera pose. For illustration purpose, model is rendered using old camera pose in frames 148 and 167. SC based matching locks the model in frame 210 and tracking starts from frame 211 onwards.

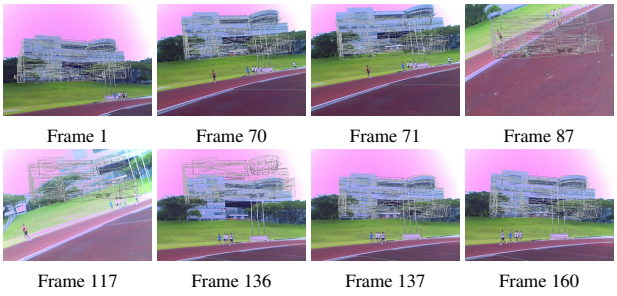


Figure 11: Results of failure recovery. Tracking starts at frame 1 and continues till frame 70. Sudden motion at frame 71 causes no updates in camera pose. For illustration purpose, model is rendered using old camera pose in frames 87 and 117. SC based matching locks the model in frame 136 and tracking starts from frame 137 onwards.

6.5 Mapping

Incremental tracking and mapping results for outdoor augmentation are illustrated for two video sequences in Fig. 12 and 13. These are closed loop tests in which the camera starts from known model-based tracking and switches over completely to model-free tracking and comes back to the starting position to illustrate the tracking accuracy.

The video of Fig. 12 consist around 492 frames and user motion is predominantly a panning type with little translation. The video of Fig. 13 is bit longer with 594 frames, and has erratic camera movements. The alignment results at the beginning and after loop closing shows the robustness of the incremental mapping algorithm. Robust tracking is achieved by frame-to-frame tracking C_{FF}, consisting of multiple features such as edges and keypoints. Edge points extracted from image silhouettes are matched with SC descriptor. For better results, please see the supplementary material of those videos.

Fig. 14 illustrates the consolidated point cloud map for the video sequence of Fig. 13. The camera has been tracked successfully in absence of any prior model data. Loop closing is successfully achieved after traversing through the loop.

7 Limitations and Further Work

OAR systems demand accurate 6-DoF pose tracking in unprepared environments. No single sensor fulfills that demand while being robust and accurate at the same time in large outdoor environments. We proposed a hybrid sensor fusion solution to address positioning, tracking and mapping issues for OAR systems. The solutions are

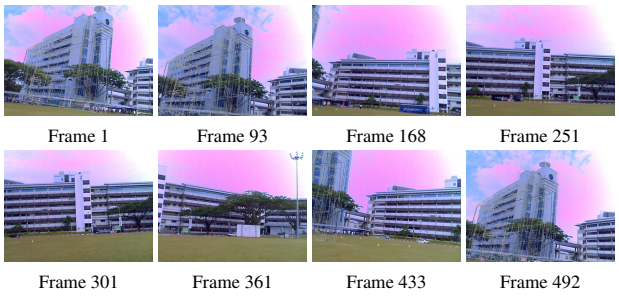


Figure 12: Results of incremental tracking and mapping approach. Tracking starts at frame 1 and comes back to initial position after 492 frames. Model augmentation at frames 1, 93 and 492 illustrates the accuracy of the proposed incremental tracking and mapping approach.

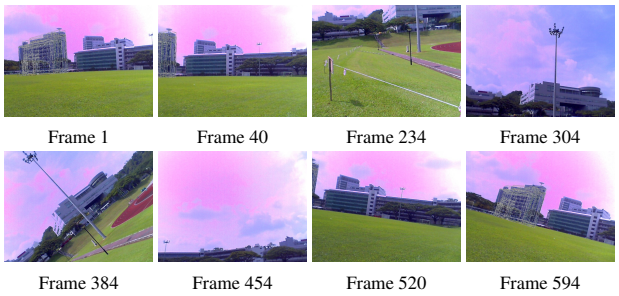


Figure 13: Results of incremental tracking and mapping approach. Tracking starts at frame 1 and comes back to initial position after 594 frames. Model augmentation at frames 1, 40 and 594 illustrates the accuracy of the proposed incremental tracking and mapping approach.

based on certain assumptions, like availability of silhouettes, which is feasible if the user is not too near the target and a silhouette can be extracted for matching and tracking. Other limitations of the proposed algorithm and further work are described below.

7.1 Positioning Failures

Our proposed one-step model-based user positioning achieves fast user positioning, however, at the cost of convergence accuracy. Shape matches obtained at the beginning between model and image silhouettes decide the fate of the final alignment result. Moreover, shape matching results based on shape context (SC) descriptors are sensitive to contour sampling and presence of extra clutter. To alleviate sensitivity of the SC based matching and local minima, repetitive model projection and shape matching can be employed. However, such an approach will give rise to initialization delays.

7.2 Tracking Failures

We achieved low-drift tracking results for longer durations by combining frame-to-frame keypoint features and model-to-frame edge tracking. Tracking results are definitely better when both these features are easily available for tracking. Nevertheless, reasonable tracking is still maintained with either of the feature matches for short durations. Hence, our system is robust to occasional loss of keypoint or edge matches. However, our system will eventually drift under persistent matching failure from either of the features.

7.3 Mapping Inadequacies

To track the user in unknown environments, we adopted an incremental tracking and mapping approach. Mapping robustness is

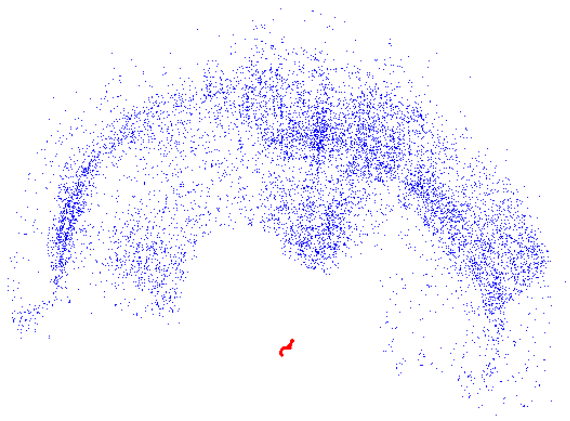


Figure 14: Point cloud demonstrates the mapped points (in blue) by incremental tracking and mapping approach for the video sequence of Fig. 13. The camera trajectory is depicted in red.

achieved by fusing different feature matches. To make it computationally light, we have neither made any attempt to refine the maps nor do we store those maps as tracking progresses. As the extent of exploration is not known a priori, the proposed approach provides adequate results. However, to take care of tracking uncertainties, robust mapping and tracking can be achieved by refining 3D maps of visible interest points as tracking progresses [5, 7].

8 Conclusion and Future Scope

In this paper we have proposed a hybrid, one-step, model-based user localization and robust tracking and mapping approaches for outdoor mixed reality applications.

The rough estimate of the camera pose available from position and orientation sensors is used to render the model. Silhouettes extracted from model and camera image are matched with the shape context descriptor. Such descriptors are ideal for OAR as it guarantees the global convergence and fast recovery from tracking failures. Further, figural continuity constraints are imposed to identify and reject outliers arising due to the presence of clutter, occlusion and partial data. The final pose is estimated by imposing box constraints. The approach is non-iterative and avoids costly rendering of the model at every iteration.

Once initialized, the algorithm switches to the combined tracking mode. Different feature matches such as keypoints and edges are appropriately weighted with respect to the current pose predicted from EKF to handle the short comings of individual trackers. Superior tracking results are obtained for longer video sequences. The combined tracking approach is even robust to small initialization errors.

The extensible mapping approach provides user tracking in unprepared environments. Such tracking is very much desirable due to ever changing dynamic outdoor conditions. Incremental mapping provides tracking capability not only for the unprepared environments but also when target object is poorly textured, such as buildings, which can lead to drifts.

The positioning, tracking and mapping framework presented in this paper for OAR assumes easy extraction of silhouette/outline from camera images, which is feasible if the user is not too near the target and presence of skyline leads to clean segmentation.

The future scope is to increase the robustness of the proposed positioning and mapping approaches. Robust user positioning can be achieved by culling the clutter in the image silhouette, which is associated with unwanted background thereby reducing the number

of outliers. Similarly, we seek to improve the tracking accuracy in unknown environments by refining the 3D maps over time.

ACKNOWLEDGEMENTS

We are very grateful to all the reviewers for providing constructive suggestions to improvise the paper. The work is funded by Singapore A*Star Project No. 062-130-0054 (WBS R-263-000-458-305): i-Explore Interactive Exploration of Cityscapes through Space and Time.

REFERENCES

[1] M. Aron, G. Simon, and M.-O. Berger. Handling uncertain sensor data in vision-based camera tracking. *In Proc. IEEE and ACM Intl. Symposium on Mixed and Augmented Reality (ISMAR'04)*, 2004.

[2] C. Arth, D. Wagner, M. Klopschitz, A. Irschara, and D. Schmalstieg. Wide area localization on mobile phones. *In Proc. IEEE and ACM Intl. Symposium on Mixed and Augmented Reality (ISMAR'09)*, 2009.

[3] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Trans. On Pattern Analysis and Machine Intelligence*, 24:509–522, 2002.

[4] S. Birchfield and S. Pundlik. Joint tracking of features and edges. *In Proc. IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR'08)*, 2008.

[5] G. Bleser, H. Wuest, and D. Stricker. Online camera pose estimation in partially known and dynamic scenes. *In Proc. IEEE and ACM Intl. Symposium on Mixed and Augmented Reality (ISMAR'06)*, 2006.

[6] P. Bouthemy. A maximum likelihood framework for determining moving edges. *IEEE Trans. On Pattern Analysis and Machine Intelligence*, 11(5):499–511, 1989.

[7] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse. Monoslam: Real-time single camera slam. *IEEE Trans. On Pattern Analysis and Machine Intelligence*, 29:1052–1067, 2007.

[8] J. Gall, B. Rosenhahn, and H.-P. Seidel. Drift-free tracking of rigid and articulated objects. *In Proc. IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR'08)*, 2008.

[9] I. Gordon and D. Lowe. Scene modelling, recognition and tracking with invariant image features. *In Proc. IEEE and ACM Intl. Symposium on Mixed and Augmented Reality (ISMAR'04)*, 2004.

[10] C. Harris and M. Stephens. A combined corner and edge detection. *In. Proc. of The Fourth Alvey Vision Conference*, pages 147–151, 1988.

[11] P. Honkamaa, S. Siltanen, J. Jappinen, C. Woodward, and O. Korkalo. Interactive outdoor mobile augmentation using markerless tracking and gps. *In Proc. IEEE Conf. on Virtual Reality (VR'07)*, 2007.

[12] Z. Hu and K. Uchimura. Fusion of vision, gps and 3d gyro data in solving camera registration problem for direct visual navigation. *Int. Journal of ITS Research*, 4(1), 2006.

[13] A. Irschara, C. Zach, J.-M. Frahm, and H. Bischof. From structure-from-motion point clouds to fast location recognition. *In Proc. IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR'09)*, 2009.

[14] B. Jiang, U. Neumann, and S. You. A robust hybrid tracking system for outdoor augmented reality. *In Proc. IEEE Conf. on Virtual Reality (VR'04)*, 2004.

[15] G. Klein and D. W. Murray. Parallel tracking and mapping for small ar workspaces. *In Proc. IEEE and ACM Intl. Symposium on Mixed and Augmented Reality (ISMAR'07)*, 2007.

[16] G. Klein and D. W. Murray. Parallel tracking and mapping on a camera phone. *In Proc. IEEE and ACM Intl. Symposium on Mixed and Augmented Reality (ISMAR'09)*, 2009.

[17] P. Moreels and P. Perona. Evaluation of feature detectors and descriptors based on 3d objects. *Int. Journal of Computer Vision*, 73:263–284, 2007.

[18] N. Paragios and R. Deriche. Geodesic active regions: A new paradigm to deal with frame partition problems in computer vision. *Journal of Vis. Comm. and Image Representation*, 13:249–268, 2002.

[19] M. Pressigout and E. Marchand. Real-time 3d model-based tracking: Combining edge and texture information. *In Proc. IEEE Intl. Conf. on Robotics and Automation (ICRA'06)*, 2006.

[20] G. Reitmayr and T. Drummond. Going out: Robust model-based tracking for outdoor augmented reality. *In Proc. IEEE and ACM Intl. Symposium on Mixed and Augmented Reality (ISMAR'06)*, pages 109–118, 2006.

[21] G. Reitmayr and T. Drummond. Initialisation for visual tracking in urban environments. *In Proc. IEEE and ACM Intl. Symposium on Mixed and Augmented Reality (ISMAR'07)*, 2007.

[22] E. Rosten and T. Drummond. Fusing points and lines for high performance tracking. *In Proc. IEEE Intl. Conf. on Computer Vision (ICCV'05)*, 2005.

[23] K. Satoh, M. Anabuki, H. Yamamoto, and H. Tamura. A hybrid registration method for outdoor augmented reality. *In Proc. IEEE and ACM Intl. Symposium on Augmented Reality (ISAR'01)*, 2001.

[24] G. Schall, D. Wagner, G. Reitmayr, E. Taichmann, M. Wieser, D. Schmalstieg, and B. Hofmann-Wellenhof. Global pose estimation using multi-sensor fusion for outdoor augmented reality. *In Proc. IEEE and ACM Intl. Symposium on Mixed and Augmented Reality (ISMAR'09)*, 2009.

[25] H. Strasdat, J. M. M. Montiel, and A. J. Davison. Real-time monocular slam: Why filter? *In Proc. IEEE Intl. Conf. on Robotics and Automation (ICRA'10)*, 2010.

[26] D.-N. Ta, W.-C. Chen, N. Gelfand, and K. Pulli. Surftrac: Efficient tracking and continuous object recognition using local feature descriptors. *In Proc. IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR'09)*, 2009.

[27] A. Thayananthan, B. Stenger, P. Torr, and R. Cipolla. Shape context and chamfer matching in cluttered scenes. *In Proc. IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR'03)*, 2003.

[28] L. Vacchetti, V. Lepetit, and P. Fua. Combining edge and texture information for real-time accrate 3d camera tracking. *In Proc. IEEE and ACM Intl. Symposium on Mixed and Augmented Reality (ISMAR'04)*, 2004.

[29] H. Wuest, F. Vial, and D. Stricker. Adaptive line tracking with multiple hypotheses for augmented reality. *In Proc. IEEE and ACM Intl. Symposium on Mixed and Augmented Reality (ISMAR'05)*, 2005.

[30] Z. Zhang. Iterative point matching for registration of free form curves and surfaces. *Int. Journal of Computer Vision*, 13:119–152, 1994.

[31] Z. Zhang. A tutorial with application to conic fitting. *Image and Vision Computing*, 15:59–76, 1997.