Realtime Motion Segmentation based Multibody Visual SLAM

Abhijit Kundu*
RRC, IIIT
Hyderabad 500032, India
abhijit.dgp@gmail.com

K. Madhava Krishna
RRC, IIIT
Hyderabad 500032, India
mkrishna@iiit.ac.in

C. V. Jawahar
CVIT, IIIT
Hyderabad 500032, India
jawahar@iiit.ac.in

ABSTRACT

In this paper, we present a practical vision based Simultaneous Localization and Mapping (SLAM) system for a highly dynamic environment. We adopt a multibody Structure from Motion (SfM) approach, which is the generalization of classical SfM to dynamic scenes with multiple rigidly moving objects. The proposed framework of multibody visual SLAM allows choosing between full 3D reconstruction or simply tracking of the moving objects, which adds flexibility to the system, for scenes containing non-rigid objects or objects having insufficient features for reconstruction. The solution demands a motion segmentation framework that can segment feature points belonging to different motions and maintain the segmentation with time. We propose a realtime incremental motion segmentation algorithm for this purpose. The motion segmentation is robust and is capable of segmenting difficult degenerate motions, where the moving objects is followed by a moving camera in the same direction. This robustness is attributed to the use of efficient geometric constraints and a probability framework which propagates the uncertainty in the system. The motion segmentation module is tightly coupled with feature tracking and visual SLAM, by exploring various feed-backs in between these modules. The integrated system can simultaneously perform realtime visual SLAM and tracking of multiple moving objects using only a single monocular camera.

1. INTRODUCTION

Both SfM from computer vision and the SLAM in mobile robotics research does the same job of estimating sensor motion and structure of an unknown static environment. The motivation behind vision based SLAM, is to estimate the 3D scene structure and camera motion from an image sequence in realtime so as to help guide robots. Vision based SLAM [3, 11, 15, 17] and SfM systems [8] have been the subject of much investigation and research. But almost all these approaches assume a static environment, containing only rigid, non-moving objects. Moving objects are treated the same way as outliers and filtered out using robust statistics like RANSAC [5]. Though this may be a feasible solution in less dynamic environments, but it soon fails as the environment becomes more and more dynamic. Also accounting for both the static and moving objects provides richer information about the environment. A robust solution to the SLAM problem in dynamic environments will expand the potential for robotic applications, especially in applications which are in close proximity to human beings and other robots. As put by [28], robots will be able to work not only for people but also with people.

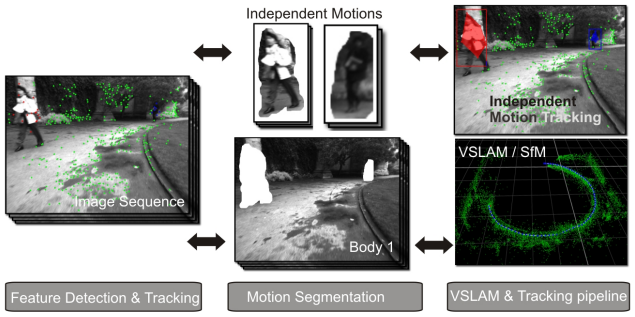

Figure 1: An Illustration of our system. Here the static background is being reconstructed, while the moving persons are being detected and tracked

The solution to the moving object detection and segmentation problem will act as a bridge between the static SLAM or SfM and its counterpart for dynamic environments. But, motion detection from a freely moving monocular camera is an ill-posed problem and a difficult task. The moving camera causes every pixel to appear moving. The apparent pixel motion of points is a combined effect of the camera motion, independent object motion, scene structure and camera perspective effects. Different views resulting from the camera motion are connected by a number of multiview geometric constraints. These constraints can be used for the motion detection task. Those inconsistent with the constraints can be labeled as moving or outliers.

The last decade saw lot of developments in the "multibody" extension [20, 21, 23, 27] to multi-view geometry. These methods are the natural generalization of the classical structure from motion theory [4, 8] to the challenging

*Corresponding author

ICVGIP '10, December 12-15, 2010, Chennai, India

case of dynamic scenes involving multiple rigid-body motions. Thus given a set of feature trajectories belonging to different independently moving bodies, multibody SfM estimates the number of moving objects in the scene, cluster the trajectories on basis of motion, and then estimate the model as in relative camera pose and 3D structure with respect to each body/object. Thus it refers to the problem of fitting multiple motion models to the scene, given a set of image feature trajectories.

By multibody visual SLAM, we indicate a realtime version of the multibody SfM. The purpose of the multibody visual SLAM is to extract as much information from the environment as possible, even those belonging to moving objects. We have taken a more practical point of view, where we choose not to reconstruct all the moving objects. This decision is motivated by the observation that foreground objects are generally small and may move rapidly and non-rigidly, which makes them very difficult for full 3D reconstruction. Moreover certain applications may just need to know the presence of moving objects, rather than its full 3D structure. The proposed framework offers the flexibility of choosing the objects that needs to be reconstructed. Objects, not chosen for reconstruction are simply tracked. Fig. 1 illustrates such a system, where the static background is chosen for reconstruction, and objects moving independently are detected and tracked over views.

The solution needs an incremental motion segmentation framework which can segment feature points belonging to different motions and maintain the segmentation with time. With every new frame it needs to verify the existing segmentation, and associate new features to one of the moving objects. We propose a realtime incremental motion segmentation algorithm for aiding multibody visual SLAM. The motion segmentation is robust and is capable of segmenting difficult degenerate motions, where the moving objects is followed by a moving camera in the same direction. Efficient geometric constraints are used in detecting these degenerate motions. We introduce a probability framework that recursively updates feature probability and takes into consideration the uncertainty in camera pose estimation. The final system integrates feature tracking, motion segmentation and 3D reconstruction by visual SLAM. We introduce several feedback paths among these modules, which enables them to mutually benefit each other. The integrated system allows simultaneous online 3D reconstruction and tracking of multiple moving objects using only a single monocular camera. A full perspective camera model is used, and we do not have any restrictive assumptions on the camera motion or environment. Unlike many of the existing works, the proposed method is online and incremental in nature and scales to arbitrarily long sequences.

In this paper, we explore in detail the motion segmentation module (Sec. 5) and its interplay with the other modules of feature tracking (Sec. 4) and visual SLAM (Sec. 6). Results of the proposed system are shown in Sec. 7 for scenes involving degenerate motions and varying number of moving objects on different datasets. Before that the previous works are detailed in Sec. 2 and Sec. 3 gives a gist overview of the whole system.

2. RELATED WORKS

The task of moving object detection and segmentation, is much easier if a stereo sensor is available, which allows additional constraints to be used for detecting independent motion [1, 2]. However the problem is very much ill-posed for monocular systems. In realtime monocular visual SLAM systems, moving objects have not yet been dealt properly. In our literature survey, we have only found three works on visual SLAM in dynamic environments: a work by Sola [26] and two other recent works of [30] and [13]. Sola [26] does an observability analysis of detecting and tracking moving objects with monocular vision. He proposes a BiCamSLAM [26] solution with stereo cameras to bypass the observability issues with mono-vision.

In [30], a 3D object tracker runs parallel with the monocular camera SLAM [3] for tracking a predefined moving object. This prevents the visual SLAM framework from incorporating moving features lying on that moving object. But the proposed approach does not perform moving object detection; so moving features apart from those lying on the tracked moving object can still corrupt the SLAM estimation. Also, they used a model based tracker, which can only track a previously modeled object with manual initialization.

The work by Migliore *et al.* [13] maintains two separate filters: a monoSLAM filter [3] with the static features and a bearing only tracker for the moving features. As concluded by Migliore *et al.* [13], the main disadvantage of their system is the inability to obtain an accurate estimate of the moving objects in the scene. This is due to the fact that they maintain separate filters for tracking each individual moving feature, without any analysis of the structure of the scene; which for e.g., can be obtained from clustering points belonging to same moving object or performing same motion. This is also the reason that they are not able to use the occlusion information of the tracked moving object, for extending the lifetime of features as in [30].

The problem of motion detection and segmentation from a moving camera has been a very active research area in computer vision community. The multiview geometric constraints used for motion detection, can be loosely divided into four categories. The first category of methods used for the task of motion detection, relies on estimating a global parametric motion model of the background. These methods [10, 19, 29] compensate camera motion by 2D homography or affine motion model and pixels consistent with the estimated model are assumed to be background and outliers to the model are defined as moving regions. However, these models are approximations which hold only for certain restricted cases of camera motion and scene structure.

The problems with 2D homography methods led to plane-parallax [9, 22, 31] based constraints. The "planar-parallax" constraints represents the scene structure by a residual displacement field termed parallax with respect to a 3D reference plane in the scene. The plane-parallax constraint was designed to detect residual motion as an after-step of 2D homography methods. They are designed to detect motion regions when dense correspondences between small baseline camera motions are available. Also, all the planar-parallax methods are ineffective when the scene cannot be approximated by a plane.

Though the planar-parallax decomposition can be used for egomotion estimation and structure, the traditional multiview geometry constrains like epipolar constraint in 2 views or trillinear constraints in 3 views and their extension to N views have proved to be much more effective in scene

understanding as in SfM or visual SLAM. This constraints are well understood and are now textbook materials [4, 8].

Most of the multibody motion segmentation research [20, 21, 23, 24, 27] has focused on theoretical and mathematical aspects of the problem. They have only been experimented on very short sequences, with either zero or very less outliers and noise-free feature trajectories. Also the high computation cost, frequent non-convergence of the solutions and highly demanding assumptions; all have prevented them from being applied to real-world sequences. Only recently Ozden *et al.* [18] discussed some of the practical issues, that comes up in multibody SfM. Though recent methods [6, 21] are more robust to outliers and noise, we are still far from doing multibody structure from motion in realtime.

3. OVERVIEW

The feature tracking module tracks existing feature points, while new features are instantiated. The purpose of the motion segmentation module is to segment these feature tracks belonging to different motion bodies, and to maintain this segmentation as new frames arrives. In the initialization step, an algebraic multibody motion segmentation algorithm is used to segment the scene into multiple rigidly moving objects. A decision is made as to which objects will be undergoing the full 3D structure and camera motion estimation. The background object is always chosen to undergo the full 3D reconstruction and camera motion estimation process. Other objects may either undergo full SfM estimation or just simply tracked, depending on the suitability for SfM estimation or application demand. On the objects, chosen for reconstruction, the standard monocular visual SLAM pipeline is used to obtain the 3D structure and camera pose relative to that object. For these objects, we compute a probabilistic likelihood that a feature is moving along or moving independently of that object. These probabilities are recursively updated as the features are tracked. Also the probabilities take care of uncertainty in pose estimation by the visual SLAM module. Features with less likelihood of fitting one model are either mismatched features arising due to tracking error or features belonging to either some other reconstructed object or one of the unmodeled independently moving objects. For the unmodeled moving objects, we use spatial proximity and motion coherence to cluster the residual feature tracks into independently moving entities.

The individual modules of feature tracking, motion segmentation and visual SLAM are tightly coupled and various feedback paths in between them are explored, which benefits each other. The motion model of a reconstructed object estimated from the visual SLAM module helps in improving the feature tracking. Relative camera pose estimates from SLAM are used by motion segmentation module to compute probabilistic model-fitness. The uncertainty in camera pose estimate is also propagated into this computation, so as to yield robust model-fitness scores. The computation of the 3D structure also helps in setting a tighter bound in the geometric constraints, which results in more accurate independent motion detection. Finally the results from the motion segmentation are fed back to the visual SLAM module. The motion segmentation prevents independent motion from corrupting the structure and motion estimation by the visual SLAM module. This also ensures a less number of outliers in the reconstruction process of a particular object. So we need less number of RANSAC iterations [5] thus resulting in improved speed in the visual SLAM module.

4. FEATURE TRACKING

Feature tracking is an important sub-module that needs to be improved for multibody visual SLAM to take place. Contrary to conventional SLAM, where the features belonging to moving objects are not important, we need to pay extra caution to feature tracking for multibody SLAM. For multibody visual SLAM to take place, we should be able to get feature tracks on the moving bodies also. This is challenging as different bodies are moving at different speeds. Also 3D reconstruction is only possible, when there are sufficient feature tracks of a particular body.

In each image, a number of salient features (FAST corners) are detected at different image pyramidal levels. Contrary to conventional visual SLAM, new features are added almost every frame. However only a subset of these, detected on certain keyframes are made into 3D points. The extra set of tracks helps in detecting independent motion. A patch is generated on these feature locations and is matched across images on the basis of zero-mean SSD scores to produce feature tracks. A number of constraints are used to improve the feature matching:

a) Adaptive Search Window: Between a pair of image, features are matched within a fixed distance (window) from its location in one image. The size and shape of this window is decided adaptively, based on the past motion of that particular body. For 3D points, whose depth has been computed from the vSLAM module, the 1D epipolar search is reduced to just around the projection of the 3D point on the image with predicted camera pose.

b) Warp matrix for patch: An affine warp is performed on the image patches to maintain view invariance from the patch's first and current observation. If the depth of a patch is unknown, only a rotation warp is made. For the image patch of the 3D points, which have been triangulated, a full affine warp is performed. This process is exactly same as the patch search procedure in Klein *et al.* [11].

c) Occlusion Constraint: Motion segmentation gives rough occlusion information, i.e. it says whether some foreground moving object is occluding some other body. This information helps in data association, particularly for features belonging to a background body, which are predicted to lie inside the convex hull created from the feature points of a foreground moving object. These occluded features are not associated, and are kept until they emerge out from occlusion.

d) Backward Match and Unicity Constraint: When a match is found, we try to match that feature backward in the original image. Matches, in which each point is the other's strongest match is kept. Enforcing unicity constraint amounts to keeping only the single strongest, out of several matches for a single feature in the other image.

5. MOTION SEGMENTATION

The input to the motion segmentation framework is feature tracks from feature tracking module, the camera relative motion in reference to each reconstructed body from the visual SLAM module, and the previous segmentation. The task of the motion segmentation module is that of model selection so as to assign these feature tracks to one of the reconstructed bodies or some unmodeled independent mo-

tion. Efficient geometric constraints are used to form a probabilistic fitness score for each reconstructed object. With each new frame, existing features are tested for model-fitness and unexplained features are assigned to one of the independently moving object. But before all this, we should initialize the motion segmentation, which is described next.

5.1 Initialization of Motion Segmentation

The initialization routine for motion segmentation and visual SLAM is somewhat different from rest of the algorithm. We make use of an algebraic two-view multibody motion segmentation algorithm of RAS [21] to segment the input set of feature trajectories into multiple moving objects. The reasons behind the choice of [21] among other algorithms is its direct non-iterative nature and faster computation time. This segmentation provides the system, the choice of motion bodies for reconstruction. For the segment chosen for reconstruction, an initial 3D structure and camera motion is computed via epipolar geometry estimation as part of static-scene visual SLAM initialization routine.

5.2 Geometric Constraints

Between any two frames, the camera motion with respect to the reconstructed body is obtained from the visual SLAM module. The geometric constraints are then estimated to detect independent motion with respect to the reconstructed body. So for the static background, all moving objects should be detected as independent motion. Epipolar constraint is the commonly used constraint that connects two views. Reprojection error or its first order approximation called Sampson error, based on the epipolar constraint is used throughout the structure and motion estimation by the visual SLAM module. Basically they measure how far a feature lies from the epipolar line induced by the corresponding feature in the other view. Though these are the gold standard cost functions for 3D reconstruction, it is not good enough for independent motion detection. If a 3D point moves along the epipolar plane formed by the two views, its projection in the image move along the epipolar line. Thus in spite of moving independently, it still satisfies the epipolar constraint. This is depicted in Fig. 2. This kind of degenerate motion is quite common in real world scenarios, e.g. camera and an object are moving in same direction as in camera mounted in car moving through a road, or camera-mounted robot following behind a moving person. To detect degenerate motion, we make use of the knowledge of camera motion and 3D structure to estimate a bound in the position of the feature along the epipolar line. We describe this as Flow Vector Bound (FVB) constraint.

5.2.1 Flow Vector Bound (FVB) Constraint:

For a general camera motion involving both rotation and translation R, t, the effect of rotation can be compensated by applying a projective transformation to the first image. This is achieved by multiplying feature points in view1 with the infinite homography $H = KRK^{-1}$ [8]. The resulting feature flow vector connecting feature position in view2 to that of the rotation compensated feature position in view1, should lie along the epipolar lines. Now assume that our camera translates by t and p_n, p_{n+1} be the image of a static point X. Here p_n is normalized as $p_n = (u, v, 1)^T$. Attaching the world frame to the camera center of the 1st view, the camera matrix for the views are $K[I|0]$ and $K[I|t]$. Also, if z is depth

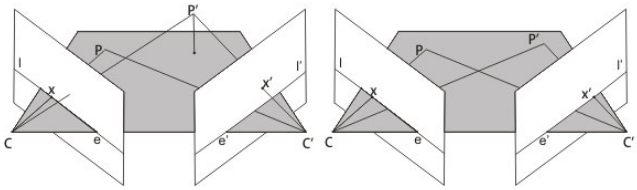

Figure 2: Left: The world point P moves non-degenerately to $P^{'}$ and hence $x^{'}$, the image of $P^{'}$ does not lie on the epipolar line corresponding to x. Right: The point P moves degenerately in the epipolar plane to $P^{'}$. Hence, despite moving, its image point lies on the epipolar line corresponding to the image of P.

of the scene point X, then inhomogeneous coordinates of X is $zK^{-1}p_n$. Now image of X in the 2nd view, $p_{n+1} = K[I|t]X$. Solving we get, [8]

$$p_{n+1} = p_n + \frac{Kt}{z} \tag{1}$$

Equation 1 describes the movement of the feature point in the image. Starting at point p_n in I_n it moves along the line defined by p_n and epipole, $e_{n+1} = Kt$. The extent of movement depends on translation t and inverse depth z. From equation 1, if we know depth z of a scene point, we can predict the position of its image along the epipolar line. In absence of any depth information, we set a possible bound in depth of a scene point as viewed from the camera. Let z_{max} and z_{min} be the upper and lower bound on possible depth of a scene point. We then find image displacements along the epipolar line, d_{min} and d_{max}, corresponding to z_{max} and z_{min} respectively. If the flow vector of a feature, does not lie between d_{min} and d_{max}, it is more likely to be an image of an independent motion.

The structure estimation from visual SLAM module helps in reducing the possible bound in depth. Instead of setting z_{max} to infinity, known depth of the background enables in setting a more tight bound, and thus better detection of degenerate motion. The depth bound is adjusted on the basis of depth distribution along the particular frustum.

The probability of satisfying flow vector bound constraint $P(FVB)$ can be computed as

$$P(FVB) = \frac{1}{1 + \left(\frac{FV - d_{mean}}{d_{range}}\right)^{2\beta}} \tag{2}$$

Here $d_{mean} = \frac{d_{min} + d_{max}}{2}$ and $d_{range} = \frac{d_{max} - d_{min}}{2}$, where d_{min} and d_{max} are the bound in image displacements. The distribution function is similar to a Butterworth band-pass filter. $P(FVB)$ has a high value if the feature lies inside the bound given by FVB constraint, and the probability falls rapidly as the feature moves away from the bound. Larger the value of β, more rapidly it falls. In our implementation, we use $\beta = 10$.

5.3 Independent Motion Probability

In this section we describe a recursive formulation based on Bayes filter to derive the probability of a projected image point of a world point being classified as stationary or dynamic. The relative pose estimation noise and image

pixel noise are bundled into a Gaussian probability distribution of the epipolar lines as derived in [8] and denoted by $EL^i = \mathcal{N}(\mu_l{}^i, \sum_l{}^i)$, where EL^i refers to the set of epipolar lines corresponding to image point i, and $\mathcal{N}(\mu_l{}^i, \sum_l{}^i)$ refers to the standard Gaussian probability distribution over this set.

Let $\mathrm{p}_n{}^i$ be the ith point in image I_n. The probability that $\mathrm{p}_n{}^i$ is classified as stationary is denoted as $P(p_n{}^i|I_n, I_{n-1}) = P_{n,s}(p^i)$ or $\mathrm{P}_{n,s}{}^i$ in short, where the suffix s signifying static. Then with Markov approximation, the recursive probability update of a point being stationary given a set of images can be derived as

$$P(p_n{}^i|I_{n+1}, I_n, I_{n-1}) = \eta_s{}^i P_{n+1,s}{}^i P_{n,s}{}^i \quad (3)$$

Here $\eta_s{}^i$ is normalization constant that ensures the probabilities sum to one.

The term $P_{n,s}{}^i$ can be modeled to incorporate the distribution of the epipolar lines EL^i. Given an image point $p_{n-1}{}^i$ in I_{n-1} and its corresponding point $p_n{}^i$ in I_n then the epipolar line that passes through $p_n{}^i$ is determined as $l_n{}^i = e_n \times p_n{}^i$. The probability distribution of the feature point being stationary or moving due to epipolar constraint is defines as

$$P_{EP,s}{}^i = \frac{1}{\sqrt{2\pi|\Sigma_l|}} exp(-\frac{1}{2}(l_n{}^i - \mu_n{}^i)^\tau \Sigma_l^{-1}(l_n{}^i - \mu_n{}^i)) \quad (4)$$

However this does not take into account the misclassification arising due to degenerate motion explained in previous sections. To overcome this, the eventual probability is fused as a combination of epipolar and flow vector bound constraints:

$$P_{n,s}{}^i = \alpha \cdot P_{EP,s}{}^i + (1-\alpha) \cdot P_{FVB,s}{}^i \quad (5)$$

where, α balances the weight of each constraint. A χ^2 test is performed to detect if the epipolar line $\mathrm{l}_n{}^i$ due to the image point is satisfying the epipolar constraint. When Epipolar constraint is not satisfied, α takes a value close to 1 rendering the FVB probability inconsequential. As the epipolar line $\mathrm{l}_n{}^i$ begins indicating a strong likelihood of satisfying epipolar constraint, the role of FVB constraint is given more importance, which can help detect the degenerate cases.

An analogous set of equations characterize the probability of an image point being dynamic, which are not delineated here due to brevity of space. In our implementation, the envelope of epipolar lines [8] is generated by a set of F matrices distributed around the mean R, t transformation between two frames as estimated by visual SLAM module. Hence a set of epipolar lines corresponding to those matrices are generated and characterized by the sample set, $EL_{ss}{}^i = \left(\hat{l}_1{}^i, \hat{l}_2{}^i .....\hat{l}_q{}^i\right)$ and the associated probability set, $P_{EL} = \left(w\hat{l}_1{}^i, w\hat{l}_2{}^i .....w\hat{l}_q{}^i\right)$ where each $\mathrm{w}\hat{l}_j{}^i$ is the probability of that line belonging to the sample set $\mathrm{EL}_{ss}{}^i$ computed through usual Gaussian procedures. Then the probability that an image point $\mathrm{p}_n{}^i$ is static is given by:

$$P_{n,s}{}^i = \sum_{j=1}^{q} \alpha_j \cdot P_{EP,\hat{l}_j{}^i}{}^S \cdot p_n{}^i + (1-\alpha_j) \cdot P_{FVB,\hat{l}_j{}^i}{}^S \cdot p_n{}^i \cdot w\hat{l}_j{}^i \quad (6)$$

where, $P_{EP,\hat{l}_j{}^i}{}^S$ and $P_{FVB,\hat{l}_j{}^i}{}^S$ are the probabilities of the point being stationary due to the respective constraints with respect to the epipolar line $\hat{l}_j{}^i$.

5.4 Clustering Unmodeled Motions

Features with high probabilities of being dynamic are either outliers or belongs to potential moving objects. Since these objects are often small, and highly dynamic, they are very hard to be reconstructed. So instead we adopt a simple move-in-unison model for them. Spatial proximity and motion coherence is used to cluster these feature tracks into independently moving entities. By motion coherence, we use the heuristic that the variance in the distance between features belonging to same object should change slowly in comparison.

6. VISUAL SLAM FRAMEWORK

The monocular visual SLAM framework is that of a standard bundle adjustment visual SLAM [11, 14, 17]. On the objects chosen for reconstruction, a 5-point algorithm [16] with RANSAC is used to estimate the initial epipolar geometry, and subsequent pose is determined with 3-point resection [7]. Some of the frames are selected as keyframes, which are used to triangulate 3D points. The set of 3D points and the corresponding keyframes are then used by the bundle adjustment process to iteratively minimize reprojection error. The bundle adjustment is initially performed over the most recent keyframes, before attempting a global optimization. Our implementation closely follows to that of [11, 14]. The system is implemented as multi-threaded processes. While one thread performs tasks like camera pose estimation, keyframe decision and addition, another backend thread optimizes this estimate by bundle adjustment.

6.1 Feedback from Motion Segmentation

However the main difference with the existing SLAM methods, is its interplay with the motion segmentation module. The motion segmentation prevents independent motion from entering the SfM computation, which could have otherwise resulted in incorrect initial SfM estimate and lead the bundle adjustment to converge to local minima. The feedback results in less number of outliers in the SfM process of a particular object. Thus the SfM estimate is more well conditioned and less number of RANSAC iterations is needed [5]. Apart from improvement in the camera motion estimate, the knowledge of the independent foreground objects coming from motion segmentation helps in the data association of the features, which are currently being occluded by that object. For the foreground independent motions, we form a convex-hull around the tracked points clustered as an independently moving entity. Existing 3D points lying inside this region is marked as not visible and is not searched for a match. This prevents 3D features from unnecessary deletion and re-initialization, just because it was occluded by an independent motion for some time.

7. EXPERIMENTAL RESULTS

The system has been tested on a number of real image datasets, with various number and type of moving entities. Details of the image sequences used for experiments are listed in Table. 1.

7.1 Moving Box Sequence

This is same sequence as used in [30]. A previously static box is being moved in front of the camera which is also moving arbitrarily. However unlike [30], our method does not

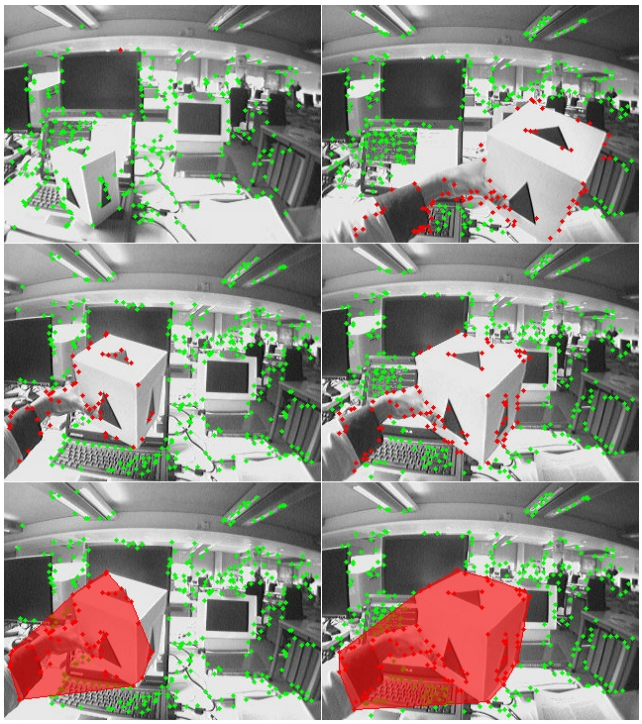

Figure 3: Results from the Moving Box Sequence

uses any 3D model, and thus can work for any previously unseen object. As shown in Fig. 3 our algorithm reliably detects the moving object just on the basis of motion constraints. The difficulty with this sequence is that the foreground moving box is nearly white and thus provides very less features. This sequence also highlights the detection of previously static moving objects. Upon detection, 3D map points lying on the moving box are deleted. The convex hull formed on the moving box is shown in red shade. This defines the occlusion mask, and corresponding actions are taken as described in Sec. 6.1. Left image of Fig. 5 shows the epipolar errors for an instance from this sequence.

Figure 4: Results from the New College Sequence.

7.2 New College Sequence

We tested our results on some dynamic parts of the New College dataset [25]. Only left of the stereo image pairs has been used. In this sequence, the camera moves along a roughly circular campus path, and three moving persons passes by the scene. Left image in Fig. 6 shows the aerial snap of the environment and the camera trajectory. Yellow denotes the part of the trajectory, when there is no independently moving body other than the static background. Green, red and blue denotes the part of trajectory where 1st, 2nd and the 3rd "moving" persons were detected. Part of the trajectory colored black denotes the time when both 2nd and 3rd moving persons are visible. Fig. 6 also shows a snap of the online map of the static background, reconstructed by the Visual SLAM framework. Fig. 4 depicts the motion segmentation results for this sequence. Fig. 5 shows an example of degenerate motion detection, as the flow vectors on the moving person almost move along epipolar lines, but they are being detected due to usage the FVB constraint. This result verifies system's performance for arbitrary camera trajectory, degenerate motion detection and changing number of moving entities.

7.3 Indoor Lab Sequence

This is an indoor sequence taken from an inexpensive hand-held camera. As the camera moves around, moving persons enter and leave the scene. Fig. 7 shows the results for this sequence. The bottom right picture in Fig. 7 shows how two spatially close independent motions is clustered correctly by the algorithm. This sequence also involves a lot of degenerate motion as the camera and the persons move in same direction. The 3D structure estimation of the background helps in setting a tighter bound in the FVB constraint. The depth bound is adjusted on the basis of depth distribution of the reconstructed background along the particular frustum, as explained in Sec. 5.2.1.

7.4 System Details

The system is implemented as threaded processes in C++. The open source libraries of TooN, OpenCV and sparse bundle adjustment [12] were used throughout the system. The run-time of the algorithm depends on lot of factors. The most significant of them are the number of bodies being reconstructed, total number of independent motions in the scene, image resolution and bundle adjustment rules. The system runs in realtime at the average of 25Hz in a standard laptop, when a single body is chosen for reconstruction. The motion segmentation module takes around 10ms for each image of 512x284 resolution and with 3 independently moving bodies.

7.5 Discussion

The results verifies that the integrated system can simultaneously perform 3D reconstruction, camera pose estimation, and tracking of multiple moving objects using only a single monocular camera, while maintaining realtime performance as listed in Table 1. Also the algorithm is online (casual) in nature as opposed to batch operation prevalent in multibody SfM literature. The proposed approach also scales to long sequences. We have shown results for degenerate motion (Fig. 5), arbitrary camera trajectory and changing number of moving entities. In Fig. 6, we demonstrated the 3D reconstruction and camera pose estimation

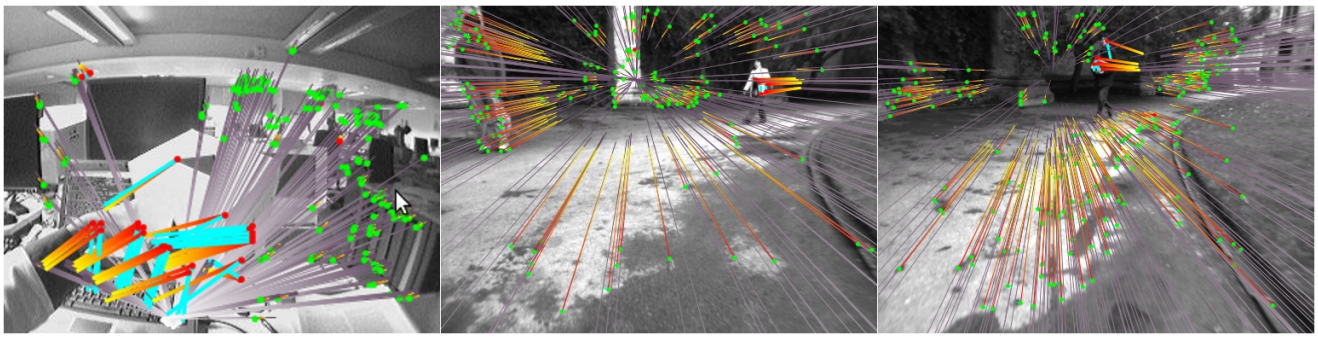

Figure 5: Epipolar lines in Grey, flow vectors after rotation compensation is shown in orange. Cyan lines show the distance to epipolar line. Features detected as independently moving are shown as red dots. Note the near-degenerate independent motion in the middle and right image. However the use of FVB constraint enables efficient detection of degenerate motion.

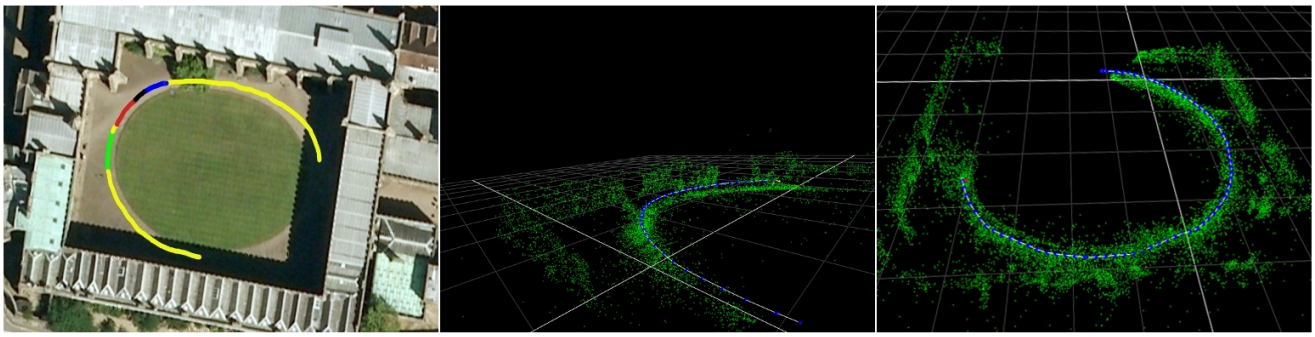

Figure 6: LEFT: Aerial map and camera trajectory. Non Yellow denotes part of the trajectory where a moving person is being detected. RIGHT: The online map being of the background, 3D structure points are in green, while white line is the camera trajectory, and the blue dots are the key-frame positions.

in reference to the static background. 3D structure points are in green, while white line is the camera trajectory, and the blue dots are the keyframe positions with respect to the background. The camera trajectory is also highlighted on the aerial map of the test environment.

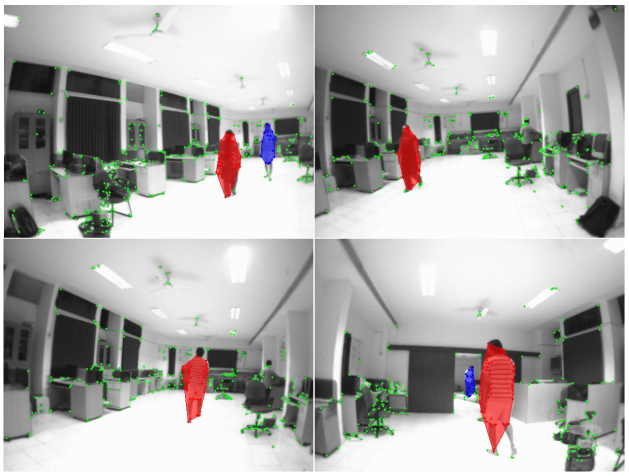

Figure 7: Results from the Indoor Lab Sequence

Table 1: Details of the datasets

Dataset	Resolution	Length	Runtime
Moving Box	320x240	718 images	30Hz
New College	512x384	1500 images	18Hz
Indoor Lab	640x480	1720 images	22Hz

8. CONCLUSIONS

This paper presents a realtime incremental motion segmentation algorithm that enables a practical multibody visual SLAM algorithm. The framework segments feature points belonging to different motions and maintain this segmentation with time. Multiview geometric constraints were explored to successfully detect various independent motion including degenerate motions. A probabilistic framework in the model of a recursive Bayes filter is developed that assigns probability to a feature being stationary or moving based on geometric constraints. Uncertainty in camera pose estimation is also propagated into this probability estimation. The different modules of motion segmentation, feature tracking and visual SLAM were integrated and we presented, how each module helps the other one. The integrated system can simultaneously perform realtime visual SLAM, and tracking of multiple moving objects using only a single monocular camera. Experiments on various real im-

age sequences shows the efficacy of the method. The work presented here can find immediate applications in various robotics applications involving dynamic scenes.

9. REFERENCES

[1] M. Agrawal, K. Konolige, and L. Iocchi. Real-time detection of independent motion using stereo. In *IEEE Workshop on Motion and Video Computing*, 2005.

[2] Z. Chen and S. Birchfield. Person following with a mobile robot using binocular feature-based tracking. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2007.

[3] A. Davison, I. Reid, N. Molton, and O. Stasse. MonoSLAM: Real-time single camera SLAM. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 29(6):1052–1067, 2007.

[4] O. Faugeras, Q. Luong, and T. Papadopoulo. *The geometry of multiple images*. MIT press, 2001.

[5] M. Fischler and R. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[6] C. G., S. Atev, and G. Lerman. Kernel Spectral Curvature Clustering (KSCC). In *ICCV'09 Workshop on Dynamical Vision*, 2009.

[7] B. Haralick, C. Lee, K. Ottenberg, and M. Nolle. Review and analysis of solutions of the three point perspective pose estimation problem. *International Journal of Computer Vision*, 13(3):331–356, 1994.

[8] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2nd edition, 2004.

[9] M. Irani and P. Anandan. A unified approach to moving object detection in 2D and 3D scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 20(6):577–589, 1998.

[10] B. Jung and G. Sukhatme. Real-time motion tracking from a mobile robot. *International Journal of Social Robotics*, pages 1–16.

[11] G. Klein and D. Murray. Parallel tracking and mapping for small AR workspaces. In *Proc. Sixth IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR)*, 2007.

[12] M. A. Lourakis and A. Argyros. SBA: A Software Package for Generic Sparse Bundle Adjustment. *ACM Trans. Math. Software*, 36(1):1–30, 2009.

[13] D. Migliore, R. Rigamonti, D. Marzorati, M. Matteucci, and D. G. Sorrenti. Avoiding moving outliers in visual SLAM by tracking moving objects. In *ICRA'09 Workshop on Safe navigation in open and dynamic environments*, 2009.

[14] E. Mouragnon, F. Dekeyser, P. Sayd, M. Lhuillier, and M. Dhome. Real time localization and 3d reconstruction. In *Computer Vision and Pattern Recognition (CVPR)*, 2006.

[15] J. Neira, A. Davison, and J. Leonard. Guest editorial, special issue in visual slam. *IEEE Transactions on Robotics*, 24(5):929–931, October 2008.

[16] D. Nister. An efficient solution to the five-point relative pose problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 26(6):756–770, 2004.

[17] D. Nister, O. Naroditsky, and J. Bergen. Visual odometry. In *Computer Vision and Pattern Recognition (CVPR)*, 2004.

[18] K. E. Ozden, K. Schindler, and L. V. Gool. Multibody structure-from-motion in practice. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 32:1134–1141, 2010.

[19] S. Pundlik and S. Birchfield. Motion segmentation at any speed. In *Proceedings of British Machine Vision Conference (BMVC)*, 2006.

[20] S. Rao, R. Tron, R. Vidal, and Y. Ma. Motion segmentation via robust subspace separation in the presence of outlying, incomplete, or corrupted trajectories. In *Computer Vision and Pattern Recognition (CVPR)*, 2008.

[21] S. Rao, A. Yang, S. Sastry, and Y. Ma. Robust Algebraic Segmentation of Mixed Rigid-Body and Planar Motions from Two Views. *International Journal of Computer Vision (IJCV)*, 2010.

[22] H. Sawhney, Y. Guo, and R. Kumar. Independent motion detection in 3D scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 22(10):1191–1199, 2000.

[23] K. Schindler and D. Suter. Two-view multibody structure-and-motion with outliers through model selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 28(6):983–995, 2006.

[24] K. Schindler, J. U, and H. Wang. Perspective n-view multibody structure-and-motion through model selection. In *European Conference on Computer Vision (ECCV)*, 2006.

[25] M. Smith, I. Baldwin, W. Churchill, R. Paul, and P. Newman. The new college vision and laser data set. *The International Journal of Robotics Research (IJRR)*, 28(5):595, 2009.

[26] J. Sola. *Towards visual localization, mapping and moving objects tracking by a mobile robot: a geometric and probabilistic approach*. PhD thesis, LAAS, Toulouse, 2007.

[27] R. Vidal, Y. Ma, S. Soatto, and S. Sastry. Two-view multibody structure from motion. *International Journal of Computer Vision (IJCV)*, 68(1):7–25, 2006.

[28] C. Wang, C. Thorpe, S. Thrun, M. Hebert, and H. Durrant-Whyte. Simultaneous localization, mapping and moving object tracking. *The International Journal of Robotics Research (IJRR)*, 26(9):889–916, 2007.

[29] J. Wang and E. Adelson. Layered representation for motion analysis. In *Computer Vision and Pattern Recognition (CVPR)*, 1993.

[30] S. Wangsiripitak and D. Murray. Avoiding moving outliers in visual SLAM by tracking moving objects. In *International Conference on Robotics and Automation (ICRA)*, 2009.

[31] C. Yuan, G. Medioni, J. Kang, and I. Cohen. Detecting motion regions in the presence of a strong parallax from a moving camera by multiview geometric constraints. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 29(9):1627–1641, 2007.