On Exploiting Per-Pixel Motion Conflicts to Extract Secondary Motions

Benzun P. Wisely Babu * Zhixin Yan† Mao Ye‡ Liu Ren§

Bosch Research and Technology Center North America
Sunnyvale, CA

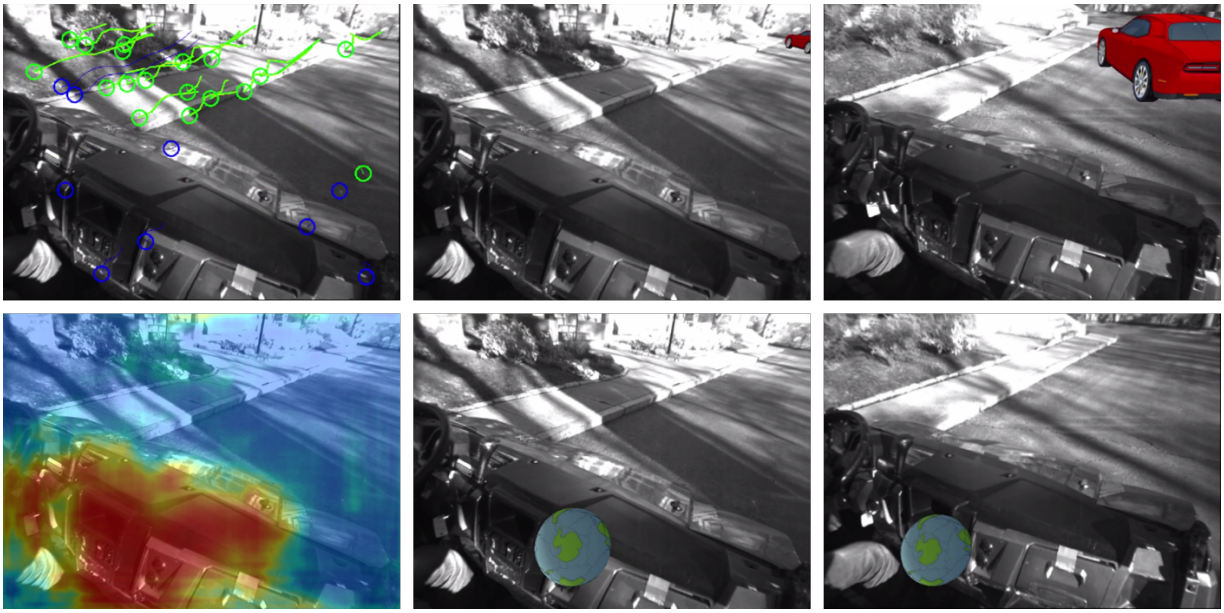

Figure 1: Our approach estimates a per-pixel motion conflict probability map (bottom left), which enables differentiation of landmarks that are associated to the primary motion and those associated to the second motions, as illustrated by the green and blue circles in the top left figure, respectively. With that, our Multi-motion MC-VIO algorithm tracks both primary and secondary motions. This enables augmentation of virtual objects attached to either of the motions. In this example, the virtual red car, attached to the primary motion appears bigger as the user moves closer (top middle and top right). By contrast, the virtual earth, attached to the secondary motion stays in the same position relative to the user (bottom middle and bottom right).

Abstract

Ubiquitous Augmented Reality requires robust localization in complex daily environments. The combination of camera and Inertial Mersurement Unit (IMU) has shown promising results for robust localization due to the complementary characteristics of the visual and inertial modalities. However, there exists many cases where the measurements from visual and inertial modalities do not provide a single consistent motion estimate thus causing disagreement on the estimated motion. Limited literature has addressed this problem associated with sensor fusion for localization. Since the disagreement is not a result of measurement noises, existing outlier rejection techniques are not suitable to address this problem. In this paper, we propose a novel approach to handle the disagreement as motion conflict with two key components. The first one is a generalized Hidden Markov Model (HMM) that formulates the tracking and management of the primary motion and the secondary motion as a single estimation problem. The second component is an epipolar constrained Deep Neural Network that generates a per-pixel motion conflict probability map. Experimental evaluations demonstrate significant improvement to the tracking accuracy in cases of strong motion conflict compared to previous state-of-the-art algorithms for localization. Moreover, as a consequence of motion tracking on the secondary maps, our solution enables augmentation of virtual content attached to secondary motions, which brings us one step closer to Ubiquitous Augmented Reality.

Keywords: Visual Inertial Odometry, Camera Pose Tracking, Motion Conflict, Sensor Fusion, Augmented Reality, Deep Neural Network

1 Introduction

Ubiquitous Augmented Reality (AR) aims to provide us a seamless, immersive user experience at anytime, anyplace. To achieve this goal, an AR system should be capable of robust localization in everyday environments and enable augmentation of digital content on the different static and moving coordinate frames in the real world. However the complexity of the environments pose major challenges:

1. It is required to robustly track the camera pose in a dynamic environment where different components in the scene undergo independent motions. For example, consider a passenger inside a moving vehicle (Fig. 2), the motion of objects seen through the window are defined with respect to the inertial

*e-mail: benzun.wisely@us.bosch.com. The first two authors are co-first authors with equal contributions.
†e-mail: zhixin.yan@us.bosch.com
‡e-mail: mao.ye2@us.bosch.com
§e-mail: liu.ren@us.bosch.com

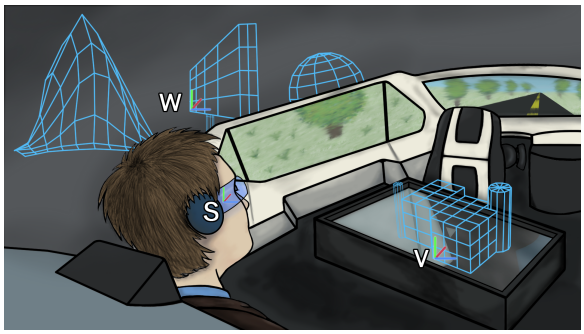

Figure 2: A passenger inside a car visualizing virtual elements attached to both the inertial coordinate frame and a local coordinate frame inside the car.

coordinate frame while the motion inside the vehicle is defined with respect to local coordinate frame. In such a dynamic environment, we need a coordinate frame aware localization to ensure robust camera pose tracking.

2. To enable the flexibility in the generation of AR content anywhere in the real world, tracking of the moving objects is needed. For example, a passenger on a moving vehicle (Fig. 2), would need to see augmented content on the inertial frame outside the vehicle such as billboards and on the local coordinate frame inside the vehicle such as the table.

Very few studies have been conducted to directly tackle these challenges, especially the second one, which is very important for Ubiquitous AR in the context of multi-sensory devices.

Over the past few decades, localization technologies (e.g., visual SLAM) have become increasingly mature [8]. It has also become widely accepted that by leveraging inertial sensors (IMU), the robustness of the localization techniques can be significantly improved. This has led to visual-inertial localization solutions (e.g., Visual-Inertial Odometry (VIO) [23], Visual-Inertial SLAM [16, 27], etc.). With the increase in computational power, commercial products based on visual-inertial localization have started to emerge on smartphones, such as ARCore and ARKit, making AR one step closer to daily usage. However, the majority of state-of-the-art visual-inertial tracking solutions assume a static environment and handle only the primary motion consistent with the inertial coordinate frame. When there are multiple motions in the scene, part of the visual measurements will disagree with the IMU measurements in terms of estimated motion, which is called *motion conflict*. The disagreement in the estimated motions by the sensors is not due to noise but rather due to the existence of primary motion consistent with the inertial frame and secondary motions consistent with local coordinate frames in the scene. Under such conditions, existing methods have very limited capability and performance.

In recent years, some attempts have been made to track the secondary motions in addition to the primary motion of the camera pose, such as [5, 38]. Nonetheless, majority of these methods rely on outlier rejection schemes that are simple and lack robustness in complex real-life scenarios, e.g., challenging illumination conditions, moving object with diverse appearance, etc. More importantly, they typically disregard the landmarks that are associated with secondary motions, resulting in the lack of capability to keep track of the secondary motions [40], which is important for Ubiquitous AR as mentioned above.

In this work, we propose a novel solution to handle motion conflicts. The key technical contributions include

1. a generalized Hidden Markov Model with time-varying states and associations to formulate the tracking of primary and secondary motions,
2. the Multi-motion MC-VIO algorithm that is able to track of the secondary motions via secondary map construction and management in addition to the primary motion,
3. a method for per-pixel motion conflict detection based on Deep Neural Networks (DNN) that leverages both visual information and inertial constraints.

Through both quantitative and qualitative experiments, we show that our solution (1) delivers significantly higher tracking accuracy and helps avoid catastrophic tracking failure in case of severe motion conflict; (2) enables augmentation of virtual contents based on secondary motions. Our novel approach brings us one step closer to Ubiquitous AR.

The remainder of the paper is organized as follows. In Section 2, we briefly summarize the related work. Next, a short background on motion conflict, notations and terminologies, are provided in Section 3. The technical details of the secondary map construction and management are presented in Section 4. It is followed by the description of epipolar constrained DNN based per-pixel motion conflict detector in Section 5. Finally, experimental evaluations and conclusions are presented in Section 6 and Section 7 respectively.

2 Related Work

There have been tremendous amounts of research on visual Simultaneous Localization and Mapping (SLAM) [3, 12]. The application of visual SLAM to enable real-time tracking in AR was initially demonstrated by Klein et. al. [20]. However, after over 30 years of research, the robustness of SLAM still needs improvement for its application to real-life conditions [8]. The use of complementary sensors such as a camera-IMU pair have shown to improve the robustness of tracking. Sensor fusion needed for camera-IMU pair has been demonstrated by either using a filtering approach such as a multi-state constraint Kalman filter [26] or by using a non-linear optimization approach such as OKVIS [23]. These sensor fusion solutions are primarily limited to tracking the motion consistent with the inertial coordinate frame and fail in dynamic conditions where the visual and inertial frames are not in agreement. The extension of SLAM to dynamic environment on devices with multiple sensors still pose many challenges. In this section, we review existing approaches for SLAM in dynamic environments.

2.1 Outlier Rejection in Tracking

Outlier rejection has been the most common approach to handle dynamic objects. The basic principle is to identify outliers by using visual cues (e.g., based on reprojection error) and then reject matches (typically sparse landmarks) that are not in agreement with the expected motion. Initial approaches such as Joint Compatibility Branch and Bound [29] and RANSAC [10] utilized hypothesis sampling to improve robustness. However, these approaches lead to erroneous estimation of motion in scenes with very few inliers. In order to overcome the limitations of RANSAC, Tan et. al. [36] introduced PARSAC that used prior adapted RANSAC to handle dynamic scenes. They also introduced online update of keyframe in dynamic environments to improve robustness of SLAM in small environments. On the contrary, our approach, performs multi-motion tracking in larger dynamic environments.

2.2 Multi motion tracking

Structure from motion has been used for multi-body tracking [15, 32, 33]. However, it was computationally expensive and was not suitable for real-time applications. Extending SLAM, approaches such as SLAMMOT [38] and SLAMIDE [5] have attempted to integrate dynamic objects tracking into SLAM. In order to extract dynamic objects, 3D motion segmentation has been demonstrated

using a number of different approaches such as normalized cuts [34], optical flow [28] and factorization [37]. Recently, Reddy et al. [30] presented a real-time multi-body tracking approach that extracted dynamic objects using motion segmentation and performed tracking of moving cars along with ego-motion estimation using factor graph optimization. However, without a fixed reference, it was difficult to determine which of the multiple motions extracted were consistent with the motion of the camera. There has been limited work to extend the visual motion segmentation to exploit the inertial information available from an IMU.

In order to integrate the dynamic objects into a single estimation problem, the Hidden Markov Model has been proposed [40]. Similar to our approach, Biswas et al. [6] presented an episodic non-Markovian localization with Variable Dynamic Graph. However, unlike our approach the episodic non-Markovian localization was demonstrated on a LIDAR based SLAM and thus did not consider measurement conflicts that exist in a multi-sensor device. We derive inspiration from Motion Conflict aware Visual Inertial Odometry (MC-VIO) algorithm [40] that considers the existence of contradictory visual-inertial measurement intervals. However, the MC-VIO algorithm was only capable of tracking the primary motion consistent with the inertial coordinate frame of the system. Besides, their approach for detection of motion conflicts was primarily based on heuristic and suffered from robustness issue in certain real-world applications.

2.3 Segmentation and Labelling

DNN, especially deep Convolutional Neural Networks(CNN), have enabled end-to-end learning for image segmentation tasks, such as semantic segmentation [2, 24], change detection [1] and instance segmentation [11]. They have outperformed the traditional methods that used hand-engineered features in most cases. These deep CNNs for image segmentation tasks usually have an encoder-decoder style architecture. The encoder extracts higher dimensional features from the original image and the decoder produces the outputs that have a similar resolution as the input. During training, the deep neural networks are able to learn how to extract visual features that contain semantic information from each type of object disregarding the change of scale, orientation, and lighting condition, given the ground truth multi-class labels. However, applying these methods, which typically only consider static visual cues (e.g., with a single image) without any temporal information, to motion conflict detection can lead to an undesirable outcome.

By contrast, we propose to leverage geometric constraints initialized by the inertial information im addition to the visual information. Our idea is partly inspired by several recent deep learning based approaches designed to solve problems that are highly correlated to geometric constraints, such as stereo matching [19, 25, 41], camera localization [17,18], object pose tracking [13] and SLAM [39,44]. In several previous work on stereo matching along this line [25,41], disparities are computed by comparing the encoded features of source image patch and target image patches from a rectified image pair. In this case, the DNNs are only used to extract high dimensional visual features, and the comparison along epipolar line is done either by another DNN or other approaches. Another way of introducing geometric constraints in DNNs is through geometric loss functions, such as reprojection loss [17, 43, 44]. In our approach, instead of applying geometric constraints outside the network or in the loss function, we create a customized layer to merge two encoded images in a way that fulfills the epipolar constraints. Our method combines the advantages of the encoder-decoder architecture mentioned above as well as the novel way of enforcing geometric constraints.

To the best of our knowledge we believe this is the first work to integrate a per-pixel classifier that detects contradictory measurements between an IMU and camera, with SLAM to perform robust multi-body tracking.

3 Motion Conflict Model

SLAM algorithms that fuse visual-inertial measurements for tracking assume complementary sensor measurements. However, we observe many conditions where this assumption is violated. Consider the case of a passenger inside a moving vehicle (Fig. 2). To display virtual content that is stationary outside the car, we need to perform tracking of the device S with respect to the inertial coordinate frame W. However, to display virtual content inside the vehicle we need to perform tracking of device S with respect to the local coordinate frame V of the car which is also moving with respect to the inertial frame. The assumption that inertial measurements are in agreement with the visual measurements is only valid in the former while violated in the latter. As described in [40], motion conflict occurs when the visual sensor disagrees with the IMU in terms of the estimated motions. We term the motion with respect to the inertial coordinate frame as primary motion and the motion with respect to the local coordinate frames as secondary motions. In this section, we extend the representation laid in [40] to motion conflict in Multi-motion scenarios.

We modelled a single estimator for both the primary and the secondary motions experienced by a multi-sensor device. A per-pixel motion conflict probability map was then used to determine the association of measurements to either primary or secondary motions. Since, feature points on dynamic objects are sparse, the per-pixel probability map reduced erroneous association of measurements to motions. By utilizing, the motion conflict probability map visual observations were classified as being consistent with the inertial coordinate frame W (primary motion) or being consistent with respect to the local frame V.

The conventional Markovian model for localization assumes a static world with a single dominant motion and complementary measurements from all the sensors. However, the static assumption is violated by the multiple independent motions that exist in the real-world scenarios [6]. Additionally, the measurements from different sensors can be inconsistent with each other [40] as they might measure different motions. Wisely Babu et al. [40] introduced a generalized Hidden Markov Model (HMM) with time varying states to handle sensor disagreements. Similarly, Biswas et al. [6] introduced the Varying Graphical Network (VGN) to handle independent motions as short-term dynamics and long-term dynamics. The HMM with time varying states assumed deterministic start and end of motion conflict interval, thus requiring a per-frame motion conflict detector. We have modelled motion conflict in a multi-motion scenario as an HMM with both varying states and associations (Fig. 4).

Similar to existing approaches, the trajectory generated by the VIO device was modelled based on the HMM. However, based on the assumptions made on the measurements we have formulated separate estimators for states outside and within the motion conflict interval. When there was only one consistent motion observed by the estimator, the state $\mathbf{X}_k$ is represented by the pose, orientation, velocity and IMU biases of the VIO device [23]. However, during a motion conflict interval, the primary and the secondary motions are represented using independent states $\mathbf{X}^W$, $\mathbf{X}^{V_n}$. Since multiple moving objects are observed by the camera in the visual frame during motion conflict interval, multiple secondary motions with corresponding local coordinate frames can be represented in the state estimator. The state associated with respect to each secondary motion was represented as $\mathbf{X}^{V_n}$, where n represented the independent moving object. For simplicity, we have assumed a single secondary motion in this work.

A Maximum à Posteriori (MAP) criterion optimization was used to estimate the states of the system. In particular, we minimize the residuals generated by the IMU observations $\hat{\mathbf{u}}$ and the visual observations $\hat{\mathbf{z}_k}$. Within a motion conflict interval, multiple MAP criterion optimizations were carried out to estimate the states of both primary and secondary motions. We used a per-pixel motion

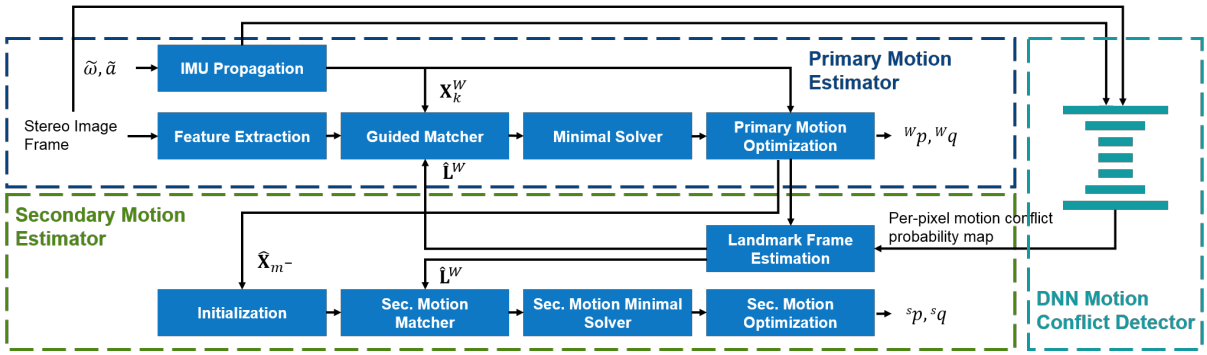

Figure 3: The block diagram of our motion conflict model that consists of a primary motion estimator, a secondary motion estimator as well as a DNN based per-pixel motion conflict detector. Details are provided in Section 3.

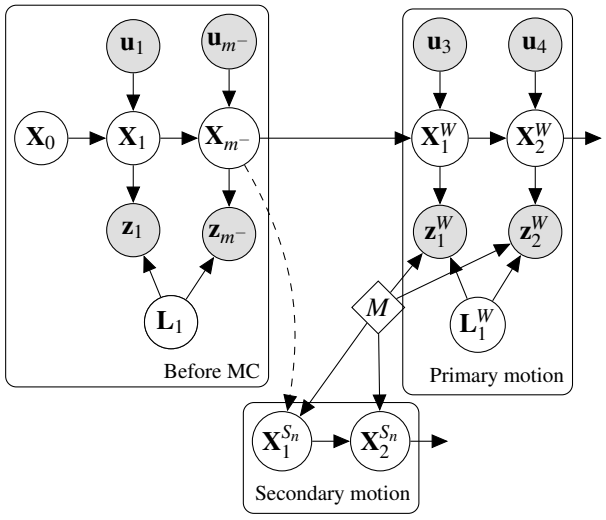

Figure 4: The Multi-motion MC-VIO was modeled as a Hidden Markov Model (HMM) with both varying states and associations. The observations are represented with gray circles and states are represented with white circles. A per-pixel motion conflict probability map M represented by white diamond is used to determine the associations.

conflict probability map M to determine the correct association of the residuals.

$$\hat{\mathbf{X}}^W = \underset{\mathbf{X}_k^W}{\operatorname{argmax}} \, \mathrm{P}(\mathbf{X}_{m^-}) \, \mathrm{P}(\mathbf{X}_{k-1}^W \mid \mathbf{X}_{m^-}) \, \mathrm{P}(\mathbf{X}^W{}_k \mid \mathbf{X}_{k-1}^W, \mathbf{u}_k, \mathbf{z}_k, M) \tag{1}$$

$$\hat{\mathbf{X}}_k^{S_n} = \underset{\mathbf{X}_k^{S_n}}{\operatorname{argmax}} \, \mathrm{P}(\mathbf{X}_{m^-}) \, \mathrm{P}(\mathbf{X}_{k-1}^{S_n} \mid \mathbf{X}_{m^-}) \, \mathrm{P}(\mathbf{X}^{S_n}{}_k \mid \mathbf{X}_{k-1}^{S_n}, \mathbf{z}_k, M) \tag{2}$$

The Markovian assumption that a state $\mathbf{X}_k$ depends only upon the input $\mathbf{u}_k$ and the previous state $\mathbf{X}_{k-1}$ is not valid as soon as a motion conflict emerges. Thus, the latest estimated state before the motion conflict, denoted as $\mathbf{X}_{\bar{m}}$, is forked into the primary motion state estimator $\mathbf{X}^W$ as well as the secondary motion state estimator $\mathbf{X}^{S_n}$. Prior to the fork, a combined state estimator that ignores the per-pixel motion conflict probability map M is used. $\mathrm{P}(\mathbf{X}_{m^-}) \, \mathrm{P}(\mathbf{X}_{k-1}^W \mid \mathbf{X}_{m^-})$ and $\mathrm{P}(\mathbf{X}_{k-1}^{S_n} \mid \mathbf{X}_{m^-})$ represent the transition probability from combined state estimator to the primary motion state estimator, and the secondary motion state estimator, respectively. In an environment where the landmarks are not stationary, the visual observations cannot be associated to the appropriate state estimator solely based on the current state $\mathbf{X}_k$ of the system. In addition to the current state, the motion assoicated with the landmark was also determined. This association was estimated using the motion conflict probability map M, which contained a per-pixel probability of disagreement with respect to the primary motion.

In summary, with the proposed HMM model, we can estimate both the primary and secondary trajectories. With these trajectories, we can render virtual objects based on either the estimated primary motion $\hat{\mathbf{X}}^W$ or the estimated secondary motions $\hat{\mathbf{X}}^{S_n}$. Notice, that compared to the previous work [40], a per-pixel motion conflict probability map M is used as part of the input to HMM in addition to the visual and inertial observations. Without M, it is not possible to associate visual measurements to primary motion and secondary motions.

4 Multi-Motion MC-VIO

In this section, we explain the Multi-motion MC-VIO which forms the backbone of our approach. A block diagram representing the different modules in our approach is presented in Figure 3.

The per-pixel motion conflict probability map M improves the estimated primary motion and the secondary motion in two ways. Firstly, it enables better outlier rejection in the primary motion estimation. Secondly, it helps determine the visual measurements that can be used for secondary motion estimation. One of the main challenges in Multi-motion MC-VIO is to keep the computational complexity of the algorithm low while performing robust secondary motion estimation with limited measurements.

The Multi-motion MC-VIO algorithm takes stereo camera images and corresponding synchronized IMU measurements as input, although the same principle applies equally to monocular systems. The motion conflict probability map is provided by our DNN-based detector presented in Section 5. The trajectory and the parameters of the VIO device are represented using state ${}^W\mathbf{X}_{0:N}$ consisting of the pose ${}^W\mathbf{p}_{WS}$, orientation $\mathbf{q}_{WS}$, velocity ${}^S\mathbf{v}$, IMU linear acceleration and rotational velocity biases $\mathbf{b}_a, \mathbf{b}_g$:

$$\mathbf{X}_k := \big[{}^W\mathbf{p}_{WS}{}^\top, \quad \mathbf{q}_{WS}{}^\top, \quad {}^S\mathbf{v}_{WS}{}^\top, \quad \mathbf{b}_g{}^\top, \quad \mathbf{b}_a{}^\top, \quad {}^W\mathbf{l}_0{}^\top, \quad \ldots, \quad {}^W\mathbf{l}_n{}^\top \big]_k^\top \in \mathbb{R}^3 \times S^3 \times \mathbb{R}^9 \times \mathbb{R}^{4n} \tag{3}$$

We have assumed that the primary motion is aligned with the inertial coordinate frame W. Hence the transition from combined state estimator to the primary motion state estimator $\mathbf{X}_k^W$ was assumed to be identity. In this work, we have assumed only a single secondary motion with respect to a local coordinate frame V. The user explicitly determines the transition from the combined state to the secondary motion state estimator $\mathbf{X}_k^{S_n}$. The assumption of

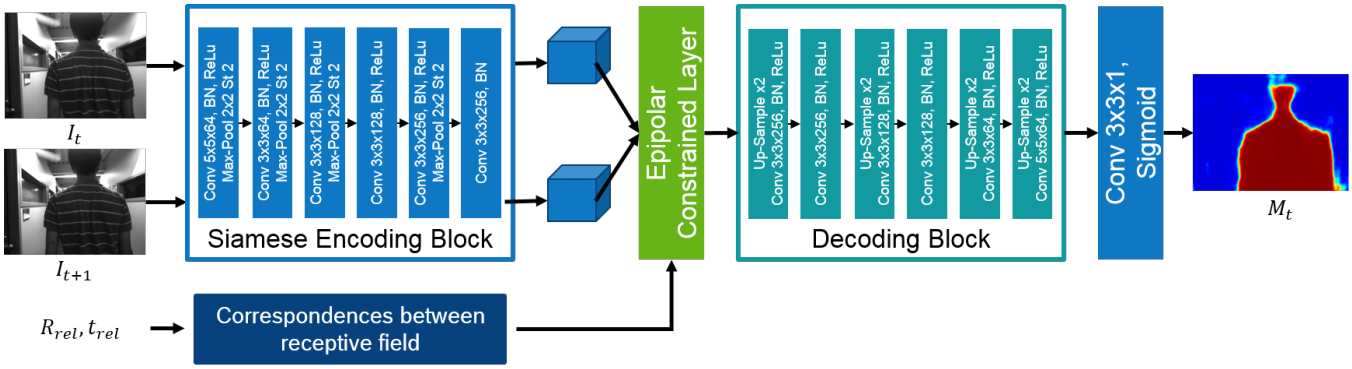

Figure 5: Our novel Deep Neural Network Architecture that includes an epipolar constrained layer to estimate per-pixel motion conflict probability map. See Section 5.1 for detailed description.

single secondary motion applies well in practice because typically the user will specify which secondary moving object (e.g., car, person, animal, etc.) is desired to be augmented on depending on the application scenario and user needs.

4.1 Primary Motion

Since an identity transformation was assumed between the combined state and the primary motion, at all times the primary motion of the system was estimated. The input to the primary state estimator consists of visual measurements $\mathbf{z}$ and IMU measurements $\mathbf{u} = [\tilde{\omega}, \tilde{\mathbf{a}}]$. The state consists of the following elements:

$$\mathbf{X}_k^W := \left[{}^W\mathbf{p}_{WS}{}^\top, \quad \mathbf{q}_{WS}{}^\top, \quad {}^S\mathbf{v}_{WS}{}^\top, \quad \mathbf{b}_g^\top, \quad \mathbf{b}_a^\top \right. \\ \left. {}^W\mathbf{l}_0{}^\top, \quad \ldots, \quad {}^W\mathbf{l}_n{}^\top \right]_k^\top \in \mathbb{R}^3 \times S^3 \times \mathbb{R}^9 \times \mathbb{R}^{4n} \tag{4}$$

A sparse feature based approach similar to [35] was used to convert stereo camera input to visual measurements $\mathbf{z}$. The FAST [31] feature detector was used to create interest points and the BRISK descriptor [22] was used for matching interest points. The previous state estimate ${}^W\mathbf{X}_{k-1}$ was propagated based on Equation 5 to estimate the a priori state ${}^W\mathbf{X}_k$.

$$\begin{aligned} {}^W\dot{\mathbf{p}} &= \mathbf{C}_{\mathbf{WB}}\, {}^B\mathbf{v} \\ \dot{\mathbf{q}}_{WB} &= \frac{1}{2}\Omega\left({}^B\tilde{\omega}_{WB} - \mathbf{b}_g\right)\mathbf{q}_{WB} \\ {}^S\dot{\mathbf{v}}_{WS} &= \left({}^S\tilde{\mathbf{a}}_{WS} - \mathbf{b}_a\right) + {}^W\mathbf{g} \\ \dot{\mathbf{b}}_g &= \mathbf{n}_{bg} \\ \dot{\mathbf{b}}_a &= -\frac{1}{\tau}\mathbf{b}_a + \mathbf{n}_{ba} \end{aligned} \tag{5}$$

The propagation equation took as input IMU measurements $(\tilde{\omega}, \tilde{\mathbf{a}})$ collected in the body frame B. The a priori state was used to guide the matcher, which generated visual correspondences between images at two different timestamps (temporal matches) and between two images at the same timestamp (static matches). These correspondences were used as visual measurements. When sufficient visual measurements of a landmark were available, triangulation was performed to initialize the landmark in the state estimator.

Since the primary motion was assumed to be aligned with the inertial coordinate frame, residuals from both the inertial measurements (Equation 6) and visual measurements (Equation 7) were used.

$$\mathbf{e}_s^k\left(\mathbf{X}_k, \mathbf{X}_{k+1}, \mathbf{z}_k, \mathbf{u}_{k-1:k}\right) = \begin{bmatrix} {}^W\hat{\mathbf{p}}_k - {}^W\mathbf{p}_k \\ 2(\hat{\mathbf{q}}_k \oplus \mathbf{q}_k^{-1}) \\ {}^S\hat{\mathbf{v}}_k - {}^S\mathbf{v}_k \\ \hat{\mathbf{b}}_{g_k} - \mathbf{b}_{g_k} \\ \hat{\mathbf{b}}_{a_k} - \mathbf{b}_{a_k} \end{bmatrix} \tag{6}$$

$$\mathbf{e}_r^{i,j,k} := \mathbf{z}^{i,j,k} - \pi_i(\mathbf{T}_{CB}\hat{\mathbf{T}}_{BW}\,{}^W\mathbf{l}_j) \tag{7}$$

Notation wise, the optimized final a posterior states were represented using $\hat{(\cdot)}$. A windowed batch optimization is performed to minimize the following energy:

$$\mathscr{J}(\mathbf{X}_k^W) := \underbrace{\sum_{k=1}^{K}\sum_{i}\sum_{j\in J(k,i)} \mathbf{e}_r^{i,k,j\top}\mathbf{W}_r\mathbf{e}_r^{i,k,j}}_{\text{reprojection error}} + \underbrace{\sum_{k=2}^{K}\mathbf{e}_s^{k\top}\mathbf{W}_s\mathbf{e}_s^k}_{\text{prediction error}} \tag{8}$$

4.2 Secondary Motion

When the user initiated tracking of secondary motion, the state $\mathbf{X}_k^{S_n}$ was estimated using the visual measurements of landmarks that were in the secondary map. The state was defined as

$$\mathbf{X}_k^{S_n} := \left[{}^V\mathbf{p}_{VB}{}^\top, \quad \mathbf{q}_{VB}{}^\top, \quad {}^V\mathbf{l}_0{}^\top, \quad \ldots, \quad {}^V\mathbf{l}_n{}^\top \right]_k^\top \in \mathbb{R}^3 \times S^3 \times \mathbb{R}^{4n} \tag{9}$$

The secondary motion estimator was initialized with the state $\hat{\mathbf{X}}_{m^-}$, which represented the last state before the start of the secondary motion tracking. To determine if a landmark was associated with the secondary map, we used the the motion conflict probability map M (see Section 5). The marginal probability based on all the visual observations $\mathbf{z}_j$ of the landmark $\mathbf{L}_i$ was given by

$$\mathrm{P}({}^V\mathbf{l}_i \mid S_n, M) = \sum_j^N \frac{\mathrm{P}(\mathbf{z}_j|\mathbf{l}_i, M)}{\mathrm{P}(\mathbf{z}_j)} \tag{10}$$

If the marginal probability for a landmark was greater than 50%, we assign the landmark to the secondary map and move all the associated residuals $e_r^{i,j,k}$ to the secondary motion estimator. We performed temporal matching of the landmarks in the secondary map with the current frame to generate additional visual measurements. The generalized P3P [21] was combined with RANSAC to estimate the pose associated with the secondary map. Finally,

bundle adjustment based on visual residuals was used to estimate the secondary motion.

$$\mathscr{J}(\mathbf{X}_k^{S_n}) := \sum_{k=1}^{K} \sum_{i} \sum_{j \in J(k,i)} \mathbf{e}_r^{i,k,j^\top} \mathbf{W}_r \mathbf{e}_r^{i,k,j} \tag{11}$$

5 DNN-Based Per-pixel Motion Conflict Detection

In the previous approach [40], per-frame motion conflict was detected to adjust the confidence of the VIO system on the visual and inertial estimates. By contrast, we aim at per-pixel detection of motion conflict to enable (1) more precise selection of landmarks for primary motion tracking and (2) tracking of secondary motions as well as augmentation based on these motions. Barnes et al. [4] proposed a multi-task deep learning approach to extract the per-pixel ephemerality mask from a single image to identify and exclude the outliers, leading to improved performance of VIO systems. However, based only on a single image, there was strong ambiguity in differentiating the primary motion from the secondary motions, especially when the secondary motions dominate the visual signals. Consequently, the estimator might behave similar to a detector for common moving objects in the training data (e.g, car, pedestrian, etc.). For example, when street parked cars are observed, there is a high chance that the network will give a positive response.

In our approach, we rely on the IMU to provide cues for the primary motion. With that, we designed an effective DNN that learnt to detect per-pixel motion conflict probability map guided by geometric constraints enforced through a novel epipolar constrained layer (Section 5.2) . In the remainder of this section, we describe the network architecture, the epipolar constraint layer, followed by a description of training and testing process.

5.1 Deep Neural Network Architecture

Our DNN is based on an encoder-decoder architecture, as shown in Figure 5. The encoder-decoder architecture has become very popular for handling per-pixel labelling, segmentation tasks [2]. The encoder block can be used to extract high dimensional features from images while the decoder block reconstruct the pixel-wise labelling result from the extracted features. Our encoding block takes two images (at time t and $t+1$) as the input, and feeds the encoded features from both images to the bottleneck layer. Here, we choose a Siamese network for feature encoding, in which the weights were shared. This choice was made based on our observation that secondary motion occurred when there were inconsistencies between the two images. To reduce the complexity of the network, we have designed a simple encoding block with five convolutional layers. Each layer was followed by a batch normalization layer and a ReLU activation except for the last one. Similar to [25], we removed the ReLU activation function from the last convolutional layer to keep the negative values in the feature before the inner-product operations described in Section 5.2. Four max pooling layers were applied between the convolutional layers.

In the bottleneck of the network, we have combined two encoded features using our customized layer as **EC** to provide epipolar geometric constraints. This layer guided the network to learn the inconsistencies between the two input images caused by secondary motions. In Section 6.2, we compared the network with a simplified variation, which does not have the epipolar constrained layer, to validate our network design choices.

The decoding block of our network included five convolutional layers, as well as four up-sampling layers, which deconvolutes the output of epipolar constrained layer **EC** and up-scales it back to the same width and height as the input images. A $3 \times 3 \times 1$ convolutional layer is applied with sigmoid activation function to generate the final output M, i.e., the motion conflict probability map with values between 0 and 1. The details such as the number of features and kernel size are provided in Figure 5.

5.2 Epipolar Constrained Layer

Our epipolar constrained layer was designed to leverage the inertial information for the purpose of motion conflict detection. The key was to utilize epipolar lines which reduce the search dimension of potential matches from 2D to 1D [14]. The design of our epipolar constrained layer is illustrated in Figure 6. For each receptive field $\mathbf{R}_{t+1}^{i,j}$ in the encoded feature of image $t+1$, we computed its inner-product with the receptive field $\{\mathbf{R}_t^{u,v}\}$ along the epipolar line $\mathbf{L} = \{u,v \mid au+bv+c=0\}$ from the other image. The results were then organized as channels of the correspondent cell $\mathbf{EC}_{i,j}$ in the output layer as follows:

$$\mathbf{EC}^{i,j} = \{\mathbf{R}_{t+1}^{i,j} \cdot \mathbf{R}_t^{u,v} \mid au+bv+c=0\} \tag{12}$$

In our implementation, instead of directly feeding inertial frames or the propagated relative pose between frame t and t+1, we feed the Epipolar Constrainted Layer with the correspondences between receptive fields of the two feature map. The main motivation was to avoid heavy computation involved in directly processing the 6DOF pose in the Epipolar Constrained Layer that was partly due to computation of fundamental matrix. To compute the correspondences between receptive fields, we first generate the Fundamental matrix F from intrinsic parameters K and the relative poses $\mathbf{R},t$ by $F = K[t]_x \mathbf{R} K^{-1}$. The intrinsic parameters K comes from camera calibration and is adjusted to the resolution of receptive fields $\mathbf{R}_t$ and $\mathbf{R}_{t+1}$. Given fundamental matrix F, the correspondences can be found along the epipolar line. With the distortion parameter calibrated, the correspondences can then be undistorted. The number of correspondences, which are the output channel of $\mathbf{EC}^{i,j}$, is fixed to $\max(H_{rf}, W_{rf})$, where W_{rf} and H_{rf} are the width and height of the encoded features, respectively. Instead of using Bresenham's line algorithm approach [7] to sample the receptive fields along the epipolar lines, we interpolate the inner-product results of the neighbor receptive fields. Zero padding is used when the epipolar line samples are outside of the boundaries.

The outcome of the epipolar constrained layer is a $H_{rf} \times W_{rf} \times \max(W_{rf}, H_{rf})$ block. For each new receptive field within the block, the feature now represents how well the receptive field $\mathbf{R}_{i,j}^{[t+1]}$ matches to the receptive field at t, provided a rough estimate of primary motion from the inertial sensor. For the regions in the image that are consistent with the primary motion, the responses in the feature $\mathbf{EC}^{i,j}$ will be high. In the case of motion conflict, the regions undergoing secondary motion will exhibit lower responses in the feature.

5.3 Training and Testing

Our network is trained using a variant of stochastic gradient descent, AdaDelta [42] which automatically adjusts the per-dimension learning rate. We minimize a pixel-wise binary cross-entropy loss:

$$\min \sum_i \sum_j -\hat{y}_{i,j} \log(y_{i,j}) + (1-\hat{y}_{i,j}) \log(1-y_{i,j}) \tag{13}$$

During the testing phase, we predict the per-pixel motion conflict probability map M for every frame and feed it to the VIO system throughout the testing sequence.

6 Experimental Results

All our experiments were carried on real-world datasets collected using a visual-inertial device consisting of a stereo camera pair and a hardware-synchronized IMU. In the rest of this section, we first describe the training data for our DNN-based motion conflict detector, and then present both quantitative and qualitative analysis of the per-pixel motion conflict probability map, as well as the resultant trajectories of the Multi-motion MC-VIO. Finally, augmentation results using the trajectories generated by the primary and secondary motions estimator have been presented.

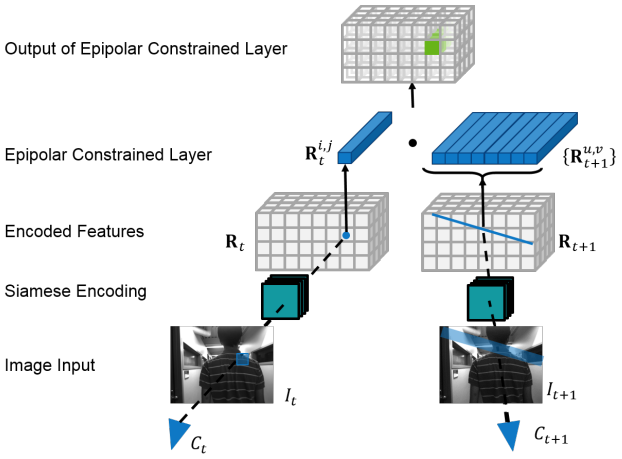

Figure 6: Illustration of our novel epipolar constrained layer and how it interacts with other layers in the network. See Section 5.2 for more details.

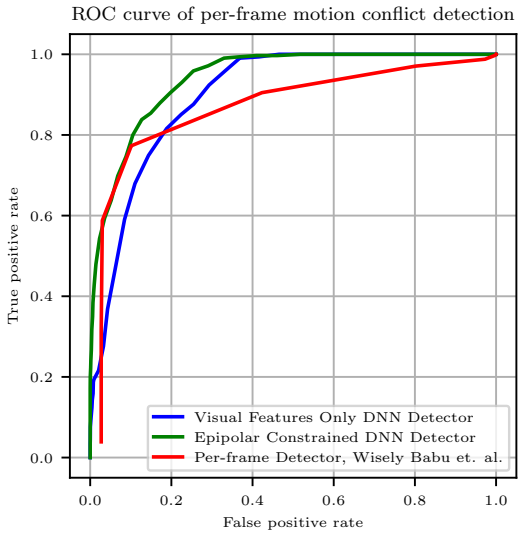

Figure 7: ROC curves on per-frame motion conflict detection. Three approaches are compared: a previous state-of-the-art [40], a baseline method (without geometric constraint) and our approach. Overall, our method achieves the best performance.

6.1 Training Data for Motion Conflict Detector

Due to the lack of existing VIO dataset that contains large variations of secondary motions, we built our training and testing sets based on the data captured in a previous work [40]. This dataset consists of five sequences, in total around 2400 seconds of visual-inertial data captured in both indoor and outdoor environments. The training set includes 80% of the three indoor sequences and one outdoor sequence (around 1623 seconds); while the testing set contains one outdoor sequence which is a driving scenario and the remaining 20% of the indoor sequences. The ground truth per-pixel motion conflict probability map is manually labelled with a semi-automatic interactive tool.

6.2 Evaluations of Motion Conflict Detector

Our motion conflict detector network was implement based on Keras [9] using an NVIDIA Tesla K40 GPU. The results were generated based on the model trained after 100 epochs with 1000 batches per epoch and a learning rate of 0.001. Although our motion conflict detection was per-pixel, we first conduct comparisons on per-frame level as no previous approach generates per-pixel result. In order to do so, the per-pixel responses were aggregated to generate an indicator for each frame. In the second part, we show the performance of our per-pixel detection result. In order to demonstrate the benefit of our epipolar constrained layer, in both parts, we have compared it with a baseline method, named *Visual features only DNN detector*, which takes only one image input without the epipolar constrained layer.

The comparison of per-frame detection results in the form of receiver operating characteristic curve (ROC) are shown in Figure 7. In particular, we compare against the previous state-of-the-art method [40], as well as the baseline method. The results indicate that just by aggregating the per-pixel responses, our network with the epipolar constrained layer can be easily converted to a robust per-frame motion conflict detector that performs better than [40]. In addition, our method largely outperforms the baseline method, which validates the benefit of our epipolar constrained layer.

For the per-pixel detection, we also plot ROC curves and compare against the baseline, as shown in Figure 8. Again, our network with epipolar constrained layer performs much better, further validating our hypothesis that the performance of motion conflict detection can be improved by enforcing geometric constraints. For qualitative evaluation, we have visualized the predicted and ground truth masks on the testing sequence in Figure 9. From the second and third row, we can see that our prediction is very close to the ground truth. In the second row, we can see that the estimation can be less accurate in black regions (lower left part) where the visual cues do not deliver very useful information. On the other aspect, our network is fairly robust in the cases where no motion conflict happens (the first and the last row). There are also certain cases where our network was not able to correctly predict the motion conflicts. Some examples are shown in Figure 10. The errors may be caused by the visual inconsistencies along the epipolar line generated from changes in light condition and reflective surfaces, e.g., sudden change in image exposure as seen in the first row.

6.3 Evaluation of Resultant Trajectories

The ultimate goal of the visual-inertial system is to deliver robust tracking. Here we provide both quantitative and qualitative analysis of the accuracy of the resultant trajectory corresponding to the primary motion. We compare the performance of the reference VIO [23], MC-VIO [40] and our Multi-motion MC-VIO on three different test sequences. Two of them are sketched in Figure 11. The first row (case 1) shows the case where a user carrying the VIO device makes three loops around a corridor while being obstructed by a moving pedestrian. The last row (case 2) corresponds to a passenger carrying the VIO device while the vehicle makes two loops around the parking lot. Some representative pictures are shown in the middle row.

For qualitative comparison, the resultant trajectory is overlaid on the corresponding 2d map in Figure 11. In both sequences, we observe that the Multi-motion VIO produces a resultant trajectory where the operator returns to the same starting point. By comparison, the reference VIO suffers from catastrophic tracking failures in both cases. The MC-VIO also has the same problem for case 1. The results clearly show that our algorithm produces more consistent trajectory than both the reference VIO and MC-VIO in environments with motion conflict.

To conduct quantitative comparison, we compare the results of these approaches against a baseline trajectory, due to lack of ground truth. The resultant trajectories that are generated by the reference VIO [23] with ground truth secondary motion labelling results are

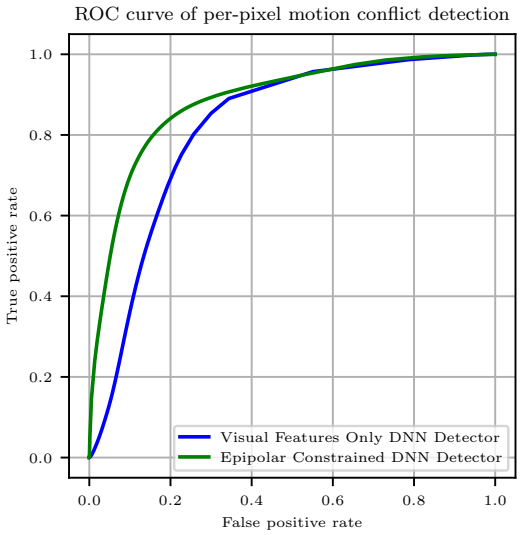

Figure 8: Quantitative comparison of per-pixel motion conflict detection between our method and the baseline approach, where no geometric constraint is used. The superior result of our method shows the effectiveness of our epipolar constrained layer.

considered as the baseline. By comparing to this baseline trajectory, how well each of the method handles the motion conflict can be quantified. Comparisons of the absolute tracking error (ATE) and relative pose error (RPE) are presented in Table 1, which demonstrates significant improvement of our approach compared to the alternatives.

6.4 Augmentation on Multiple Motions

As mentioned in Section 1, in Ubiquitous AR, the user should be able to augment digital content on almost anywhere in the real world. To showcase that our method is one step further in enabling such feature, we present augmentation results based on the resultant trajectory generated by our Multi-motion MC-VIO in Figure 12. In this example sequence, the passenger carries the VIO devices while the vehicle is in motion. We augment a red virtual car attached to the primary motion of the VIO device and a virtual earth rendered inside the vehicle attached to the secondary motion. As the car moves, the red virtual car that is fixed to the world frame will become closer to the camera; while the virtual earth will remain roughly in the same size and move consistently with the car. Our results in Figure 12 show visual effects that are consistent with this expectation. We can conclude that our Multi-motion MC-VIO enables successful tracking of both the primary and the secondary motions in real-world conditions.

Table 1: Evaluation of Multi-motion MC-VIO (Ours) in comparison with two state-of-the-art methods on datasets with clear motion conflict, using the baseline trajectory as ground truth.

Dataset	ATE $[m]$			RPE $[m/s]$		
	OKVIS	MC-VIO	Ours	OKVIS	MC-VIO	Ours
Seq1	4.979	4.681	**0.568**	0.052	0.047	**0.018**
Seq2	4.873	6.391	**0.142**	0.052	0.051	**0.010**
Seq3	27.22	2.740	**1.932**	0.252	0.065	**0.029**
mean	12.35	4.604	**0.881**	0.119	0.054	**0.019**
std	10.51	1.492	**0.763**	0.094	**0.008**	**0.008**

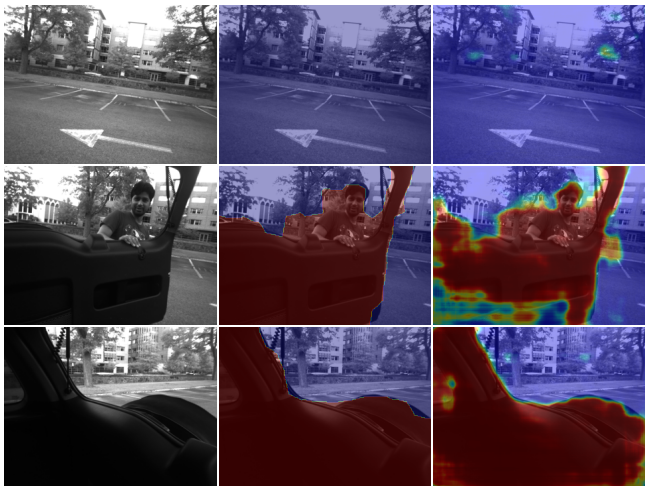

Figure 9: Qualitative evaluation of our estimated motion conflict probability map (right column) against the ground truth (middle column), where we overlay the mask on the original image (left column). For our results, the jet color scheme is used for visualization. The four rows represent different timestamps of a sequence, where motion conflict happens in the second and the third rows, but not in the first and the last rows. See Section 6.2 for analysis.

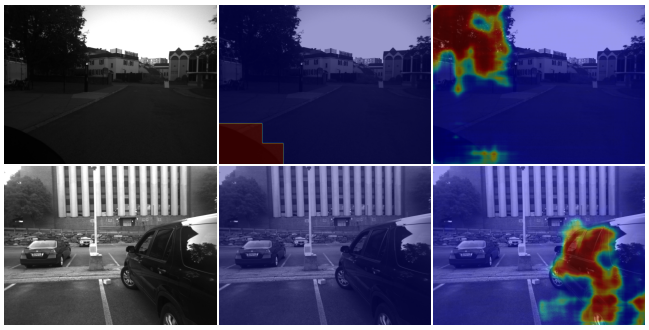

Figure 10: Examples where motion conflict detection failed. From left to right: captured image, ground truth motion conflict labelling, predicted motion conflict probability map. The error might be caused by inconsistency of visual information between two input images, which is also a common challenge for other vision problems including stereo matching.

7 Conclusion

In the paper, we aim to address the motion conflict problem during visual-inertial tracking and mapping for Ubiquitous AR. We proposed a novel HMM-based motion conflict model which can properly formulate the state and association changes caused by secondary motions during VIO. Based on the model, we developed the Multi-motion MC-VIO algorithm that includes a novel DNN-based per-pixel motion conflict detector. The key to our detector is a novel epipolar constrained layer that enforces geometric constraints based on a rough estimation of primary motion using IMU measurements. Given the motion conflict probability map, our Multi-motion MC-VIO algorithm is able to track and manage the secondary motion along with the primary motion. Our experimental results demonstrate that our Multi-motion MC-VIO significantly outperforms the previous state-of-the-art algorithm for localization on datasets that include severe motion conflicts. Additionally, with the tracked secondary map, our solution enables augmentation of virtual objects to

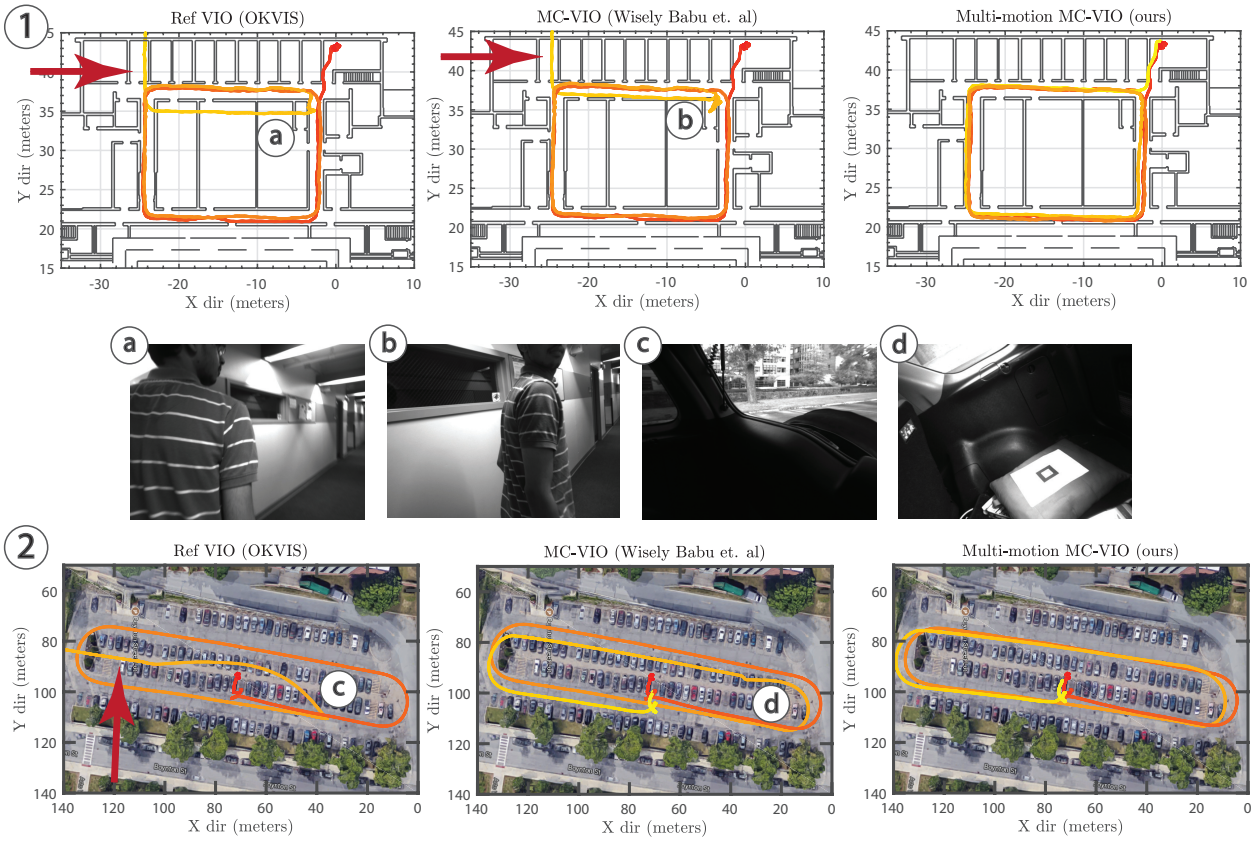

Figure 11: Resultant primary motion trajectories comparing OKVIS [23], MC-VIO [40] and our Multi-motion MC-VIO on indoor (1) and outdoor datasets (2) are presented. Representative images during motion conflict intervals that caused failures of OKVIS algorithm are presented in a and b. Similarly the representative images for MC-VIO algorithm with per-frame motion conflict are presented in c and d. Their corresponding locations on the trajectory are also marked with a, b, c, and d, respectively. Apparently, our Multi-motion MC-VIO based on the per-pixel motion conflict detection achieves the best performance. The OKVIS algorithm suffers from catastrophic failure in both cases, so does MC-VIO on the first case (see the parts of trajectories pointed with red arrows). For the second case, the MC-VIO algorithm shows much larger error (the two loops do not overlap). In both cases, our algorithm delivers much better trajectories.

Figure 12: Augmentation based on the primary and secondary motions. The top row contains the features tracked for primary motion estimation (green) and features tracked for secondary motion estimation (blue). The majority of the features are correctly classified based on our motion conflict detection results. The middle row contains a virtual red car rendered based on the primary motion (i.e., parked on the road). The bottom row contains a virtual earth rendered on the secondary motion (i.e., attached inside the car). Intuitively, as the car moves forward, the virtual car should appear larger; while the virtual earth should move along with the car and stay in the same size. Our visual rendering matches this expectation, demonstrating the capability of our solution to track both primary and secondary motions.

both primary motion and secondary motion without any high-level semantic cues, making it one step closer to the ultimate goal of Ubiquitous AR.

There are certain limitations to our approach. The secondary motion estimator is affected when there is a lack of distinct visual features, leading to degradation of the resultant trajectory. Moreover, our implementation does not contain loop closures that can help improve the accuracy of the system. So far, our DNN is only trained and tested in a relatively small dataset (in terms of number of images, variations of motion conflicts, etc.). In the future, more extensive evaluations need to be conducted to analyze potential problems (e.g., over-fitting) and better understand the limitation of the method. Additionally, the epipolar constrained layer has limitations when dealing with reflective materials and drastic exposure changes. Nonetheless, with our current solution that has demonstrated major improvements over the existing work, we hope to inspire more studies along this line towards Ubiquitous AR.

REFERENCES

[1] P. F. Alcantarilla, S. Stent, G. Ros, R. Arroyo, and R. Gherardi. Street-view change detection with deconvolutional networks. In *Proceedings of Robotics: Science and Systems*. AnnArbor, Michigan, June 2016. doi: 10.15607/RSS.2016.XII.044

[2] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495, 2017.

[3] T. Bailey and H. Durrant-Whyte. Simultaneous localization and mapping: Part II. *IEEE Robotics Automation Magazine*, 13(3):108 – 117, 2006.

[4] D. Barnes, W. Maddern, G. Pascoe, and I. Posner. Driven to Distraction: Self-Supervised Distractor Learning for Robust Monocular Visual Odometry in Urban Environments. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. Brisbane, May 2018.

[5] C. Bibby and I. Reid. Simultaneous Localisation and Mapping in Dynamic Environments (SLAMIDE) with Reversible Data Association. In *Proceedings of Robotics: Science and Systems*. Atlanta, GA, USA, June 2007. doi: 10.15607/RSS.2007.III.014

[6] J. Biswas and M. Veloso. Episodic non-markov localization: Reasoning about short-term and long-term features. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3969–3974, May 2014. doi: 10.1109/ICRA.2014.6907435

[7] J. E. Bresenham. Algorithm for computer control of a digital plotter. *IBM Systems journal*, 4(1):25–30, 1965.

[8] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. D. Reid, and J. J. Leonard. Simultaneous localization and mapping: Present, future, and the robust-perception age. *IEEE Transactions on Robotics*, 32(6):1309–1332, Dec 2016.

[9] F. Chollet et al. Keras. `https://github.com/keras-team/keras`, 2015.

[10] J. Civera, O. G. Grasa, A. J. Davison, and J. M. M. Montiel. 1-Point RANSAC for extended kalman filtering: Application to real-time structure from motion and visual odometry. *Journal of Field Robotics*, 27(5):609631, 2010.

[11] J. Dai, K. He, and J. Sun. Instance-aware semantic segmentation via multi-task network cascades. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3150–3158, 2016.

[12] H. Durrant-Whyte and T. Bailey. Simultaneous localization and mapping: Part I. *IEEE Robotics Automation Magazine*, 13(2):99 – 110, 2006.

[13] M. Garon and J.-F. Lalonde. Deep 6-DOF tracking. *IEEE Transactions on Visualization and Computer Graphics*, 23(11), November 2017.

[14] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second ed., 2004.

[15] C. Jiang, D. P. Paudel, Y. Fougerolle, D. Fofi, and C. Demonceaux. Static-map and dynamic object reconstruction in outdoor scenes using 3-d motion segmentation. *IEEE Robotics and Automation Letters*, 1(1):324–331, Jan 2016. doi: 10.1109/LRA.2016.2517207

[16] E. S. Jones and S. Soatto. Visual-inertial navigation, mapping and localization: A scalable real-time causal approach. *The International Journal of Robotics Research*, 30(4):407–430, 2011. doi: 10.1177/0278364910388963

[17] A. Kendall and R. Cipolla. Geometric loss functions for camera pose regression with deep learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[18] A. Kendall, M. Grimes, and R. Cipolla. Posenet: A convolutional network for real-time 6-dof camera relocalization. In *Proceedings of the International Conference on Computer Vision (ICCV)*, pp. 2938–2946. IEEE, 2015.

[19] A. Kendall, H. Martirosyan, S. Dasgupta, P. Henry, R. Kennedy, A. Bachrach, and A. Bry. End-to-end learning of geometry and context for deep stereo regression. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2017.

[20] G. Klein and D. Murray. Parallel Tracking and Mapping for Small AR Workspaces. In *Proc. IEEE and ACM International Symposium on Mixed and Augumented Reality*, pp. 225–234. Nara, Japan, Nov 2007.

[21] L. Kneip, D. Scaramuzza, and R. Siegwart. A novel parametrization of the perspective-three-point problem for a direct computation of absolute camera position and orientation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2969–2976, June 2011. doi: 10.1109/CVPR.2011.5995464

[22] S. Leutenegger, M. Chli, and R. Y. Siegwart. Brisk: Binary robust invariant scalable keypoints. In *In Proceedings of the International Conference on Computer Vision (ICCV)*, pp. 2548–2555, Nov 2011. doi: 10.1109/ICCV.2011.6126542

[23] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale. Keyframe-based visual-inertial odometry using nonlinear optimization. *The International Journal of Robotics Research*, 34(3):314–334, 2015.

[24] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431–3440, 2015.

[25] W. Luo, A. G. Schwing, and R. Urtasun. Efficient deep learning for stereo matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5695–5703. Las Vegas, June 2016.

[26] A. I. Mourikis and S. I. Roumeliotis. A multi-state constraint kalman filter for vision-aided inertial navigation. In *Proc. IEEE Int. Conf. International Conference on Robotics and Automation*, pp. 3565–3572. Hamburg, Germany, April 2007.

[27] R. Mur-Artal and J. D. Tardós. Visual-inertial monocular slam with map reuse. *IEEE Robotics and Automation Letters*, 2(2):796–803, 2017.

[28] M. Narayana, A. Hanson, and E. Learned-Miller. Coherent motion segmentation in moving camera videos using optical flow orientations. In *Proceedings of the 2013 IEEE International Conference on Computer Vision*, ICCV '13, pp. 1577–1584. IEEE Computer Society, Washington, DC, USA, 2013. doi: 10.1109/ICCV.2013.199

[29] J. Neira and J. D. Tardos. Data association in stochastic mapping using the joint compatibility test. *IEEE Transactions on Robotics and Automation*, 17(6):890–897, Dec 2001.

[30] N. D. Reddy, I. Abbasnejad, S. Reddy, A. K. Mondal, and V. Devalla. Incremental real-time multibody VSLAM with trajectory optimization using stereo camera. In *Proc. IEEE Int. Conf. Intelligent Robots and Systems*. Daejeon, Korea, Oct 2016.

[31] E. Rosten and T. Drummond. Fusing points and lines for high performance tracking. In *Proceedings of the International Conference on Computer Vision (ICCV)*, vol. 2, pp. 1508–1515 Vol. 2, Oct 2005. doi: 10.1109/ICCV.2005.104

[32] A. Roussos, C. Russell, R. Garg, and L. Agapito. Dense multibody motion estimation and reconstruction from a handheld camera. In *2012 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 31–40, Nov 2012. doi: 10.1109/ISMAR.2012.6402535

[33] R. Sabzevari and D. Scaramuzza. Multi-body motion estimation from monocular vehicle-mounted cameras. *IEEE Transactions on Robotics*, 32(3):638–651, June 2016. doi: 10.1109/TRO.2016.2552548

[34] J. Shi and J. Malik. Motion segmentation and tracking using normalized cuts. In *Sixth International Conference on Computer Vision (IEEE Cat.*

No.98CH36271), pp. 1154–1160, Jan 1998. doi: 10.1109/ICCV.1998.710861

[35] H. Strasdat, A. J. Davison, J. M. M. Montiel, and K. Konolige. Double window optimisation for constant time visual slam. In *2011 International Conference on Computer Vision*, pp. 2352–2359, Nov 2011.

[36] W. Tan, H. Liu, Z. Dong, G. Zhang, and H. Bao. Robust monocular slam in dynamic environments. In *2013 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 209–218, Oct 2013. doi: 10.1109/ISMAR.2013.6671781

[37] R. Vidal and Y. Ma. A unified algebraic approach to 2-d and 3-d motion segmentation. In T. Pajdla and J. Matas, eds., *Computer Vision - ECCV 2004*, pp. 1–15. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004.

[38] C.-C. Wang and C. Thorpe. Simultaneous localization and mapping with detection and tracking of moving objects. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. Washington, DC, May 2002.

[39] S. Wang, R. Clark, H. Wen, and N. Trigoni. Deepvo: Towards end-to-end visual odometry with deep recurrent convolutional neural networks. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2043–2050. IEEE, 2017.

[40] B. P. Wisely Babu, D. Cyganski, J. Duckworth, and S. Kim. Detection and Resolution of Motion Conflict in Visual Inertial Odometry. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. Brisbane, May 2018.

[41] J. Zbontar and Y. LeCun. Stereo matching by training a convolutional neural network to compare image patches. *Journal of Machine Learning Research*, 17(1-32):2, 2016.

[42] M. D. Zeiler. Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*, 2012.

[43] Y. Zhong, Y. Dai, and H. Li. Self-supervised learning for stereo matching with self-improving ability. *arXiv preprint arXiv:1709.00930*, 2017.

[44] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe. Unsupervised learning of depth and ego-motion from video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.