Visual-Inertial Source Localization for Co-Robot Rendezvous

Xi Sun

CMU-RI-TR-20-15

May 19, 2020

The Robotics Institute
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA

Thesis Committee:
Prof. Kris Kitani, *chair*, CMU
Prof. David Held, CMU
Xiaofang Wang, CMU

Submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Robotics.

Abstract

We aim to enable robots to visually localize a target person through the aid of an additional sensing modality – the target person's 3D inertial measurements. The need for such technology may arise when a robot is to meet a person in a crowd for the first time or when an autonomous vehicle must rendezvous with a rider amongst a crowd without knowing the appearance of the person in advance. A person's inertial information can be measured with a wearable device such as a smart-phone and can be shared selectively with an autonomous system during the rendezvous. We describe a method for learning a visual-inertial feature space in which the motion of a person in video can be easily matched to motion measured by a wearable inertial measurement unit (IMU). The transformation of the two modalities into the joint feature space is learned through the use of a contrastive loss which forces inertial motion features and video motion features generated by the same person to lie close in the representational feature space. To validate our approach, we compose a dataset of over 60,000 video segments of moving people along with wearable IMU data. Our experiments show that our proposed algorithm is able to accurately identify a target person in a realistic multi-person scenario with 72.4% accuracy using only 5 seconds of IMU data and video.

Acknowledgments

I would like to thank my advisor, Professor Kris Kitani, for providing me with the opportunity to work on this project. His advice and guidance have been invaluable during my research and will continue to play an important role on my path moving forward.

I would like to thank Xinshuo Weng for further helping me improve my research skills and the great support over a year. I am grateful to all the members in KLab for their help with the data collection. I would like to thank Professor David Held and Xiaofang Wang for providing insightful feedback as my thesis committee. I also appreciate the sponsorship from Highmark on this project.

Contents

When this dissertation is viewed as a PDF, the page header is a link to this Table of Contents.

List of Figures

List of Tables

Chapter 1

Introduction

Person localization for a rendezvous is crucial in real-world applications such as assistive robots [14, 20] and autonomous driving [3, 4, 15, 16, 19, 21, 28, 29, 30, 31, 32, 33, 35, 36]. Consider the scenario where an autonomous vehicle rendezvous with it's user for the first time. How does the autonomous vehicle localize the user without any information about what the user looks like? In this work, we consider the possibility of using the user's inertial measurement unit (IMU) data collected by her smartphone as a unique descriptor of the user's motion, which can be then used by the autonomous vehicle to localize the user with a dashboard camera.

Prior work on person localization often utilizes visual-visual feature matching, assuming that the target person's appearance information is known in advance. However, this assumption may not always hold as it requires a data capture process prior to the rendezvous. To deal with the situation where the target person's appearance information is not available, we must rely on other sensor that can capture target person's information in the wild. We choose to use the 3D inertial sensor as the 3D inertial measurement describes the user's motion and can be matched with the visual motion information collected by the dash camera for person localization. Also, the user's 3D inertial measurement can be easily obtained because modern smart wearable devices such as smart-phone and smart-watch are often equipped with an inertial sensor. Moreover, due to its low dimensionality compared to visual data, we can transmit the inertial measurement to the autonomous vehicle in real time at a low cost.

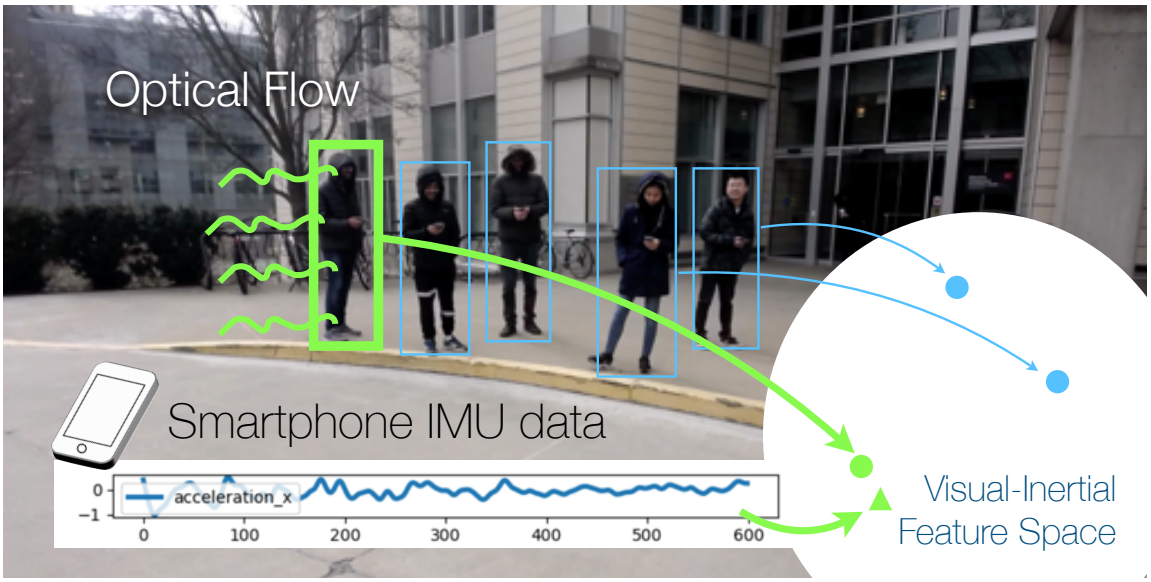

Figure 1.1: Our visual-inertial feature transformer maps IMU motion and image motion from the same person to a similar location in the feature space.

Our approach is based on visual-inertial feature matching. Specifically, we first obtain the visual motion information from the dashboard camera by computing the optical flow [5] for a fixed time window. In the meantime, we obtain the motion information in 3D space measured by the inertial measurement unit (IMU) for the same time window. Since directly transforming the local 3D motion measurements and the 2D motion in the camera frame into same world coordinates is difficult and requires calibration of a fixed camera, we propose to learn a feature transformer based on LSTM [11] and Multi-Layer Perceptron (MLP) that can map the motion information from visual and inertial modalities into a joint feature space. The visual and inertial features are optimized using a contrastive loss [23] so that the learned features of the same person lie close in the joint feature space.

As there is no existing dataset suitable for training our feature transformer for person localization, we collect a new visual-inertial dataset containing time-synchronized video and inertial data. Our dataset has over 60,000 video segments of moving people along with their corresponding IMU data. The IMU data is collected by the smartphones held in people's hands. Different from existing visual-inertial datasets which often rigidly attach the inertial sensor on people's back [9] or body

limbs [13, 26, 27], we let people hold smartphones in their hands naturally to mimic the real-world scenarios. As a result, our dataset is more realistic but challenging as the location of the inertial sensor is more flexible and the motion of the inertial sensor might not always align with the motion of people's back or limbs.

To validate our approach, we evaluate it on the test split of our visual-inertial dataset. Our experiments show that our approach is able to accurately identify a target person with 72.4% accuracy using only 5 seconds of IMU and video data. To summarize, our contributions are as follows:

1. **A new task, namely visual-inertial person localization**, which aims to localize the target without requiring the appearance information of the target in advance;
2. **A new large visual-inertial dataset**, which is collected in the wild with multiple persons without fixed attachment of the inertial sensor to each person's body;
3. **An effective approach for the proposed task**, also being the first learning-based approach for the task and outperforming competitive baselines we devised from state-of-the-art techniques.

Chapter 2

Related Work

2.1 Visual-Inertial Person Localization

To the best of our knowledge, [9] is the only work that attempted matching between visual and inertial data for person localization. First, [9] employs a visual heading network to predict person's 3D orientation with respect to the camera from a single image. Then, they match the person's 3D orientation predicted from the image with the orientation integrated from angular velocity obtained from the inertial sensor to generate image-based person predictions. To rigidly align the orientation of the inertial sensor with the person's body orientation and make the orientation prediction problem easier, [9] attaches the inertial sensor on the back of the target person. This makes [9] not applicable in the real world scenarios where the inertial sensor can be flexible. Additionally, [9] employs velocity matching between inertial and visual data to formulate trajectories of the previously generated imaged-based predictions. Specifically, the 3D foot position of the person is estimated from an image, which is then used to compute the 3D velocity of the target person given a pair of images. Meanwhile, the 3D velocity is also estimated by integrating the linear acceleration from the inertial data, which can be used to match with the 3D velocity computed from the visual data. Different from [9] which employs hand-crafted inertial features (*i.e.* orientation and velocity obtained by integration) to match with the visual data, our proposed method learns to transform visual and inertial data into a joint feature space for matching. Also, our proposed method is more useful in real world scenarios

as we do not restrict the placement of the inertial sensor.

2.2 Visual-Inertial Dataset

Although visual-inertial person localization is under-explored in prior work, there are existing visual-inertial datasets collected for other vision tasks. The CMU Multi-Modal Activity Database [8] aims to understand cooking and food preparation activities. They rigidly attach multiple IMU sensors on person's body to collect the inertial data. In the meantime, video data is also collected from multiple viewpoints. The Total Capture Dataset [24] is designed for human pose estimation. Similarly, [24] contains synchronized multi-view video and IMU data with the inertial sensor attached to the human body. However, both [8] and [24] are not suitable for person localization as 1) they only collect data for one person at a time, 2) the location of the inertial sensor is fixed, and 3) the data is collected in the indoor setting. Different from existing datasets, we collect a new visual-inertial dataset with multiple persons outside and the location of the inertial sensor flexible, in order to mimic the real-world autonomous driving pick-up scenario.

2.3 Visual Person Localization

Depart from the visual-inertial person localization, prior work has investigated person localization using only visual data with the re-identification technique. The common approach is to first obtain the feature embedding from two sources of visual data (one from an unknown query person and the other from a pre-built database containing information of the target person), and then perform classification to identify if the query person is the target person. Once the target person is successfully identified, the localization is solved. To obtain effective visual embedding for identification and localization, prior work focuses on image-based [6, 18, 37] and video-based [17] methods for feature learning. However, visual person localization methods are only applicable when the pre-built database containing the information of the target person is available. In other words, if we do not have the target person's information in advance, we cannot solve the localization problem with only visual information but

need the aid of an additional sensor. In this paper, we investigate the possibility of using the user's inertial data for localization.

2.4 Visual-Inertial Human Pose Estimation

In addition to person localization, prior work has investigated using inertial and visual data for other computer vision applications such as human pose estimation. [27] proposes a Video Inertial Poser (VIP) to perform accurate 3D human motion capture using 6 to 17 IMUs attached at the person's body limbs and a single hand-held moving phone camera. Specifically, they first obtain a set of initial 3D poses from the IMU data and also obtain 2D human pose keypoints on the 2D video. Then, they associate the 2D poses and 3D poses and obtain the globally consistent assignment by jointly optimizing the cost over camera pose, people's heading angle and 3D poses. Although the task is different from ours, this work also aims to find the correlation between the 2D and 3D features from visual and inertial modalities. However, again, this work relies on a large number of IMU sensors and does not apply to the practical scenario in our problem setup.

Chapter 3

Method

Given a video with multiple people standing or walking, and the IMU readings from a smartphone carried by a person in the scene, our goal is to identify which person in the video the IMU data belongs to, as shown in Fig. 1.1. As described above, we aim to learn a joint visual-inertial feature space in which the visual and inertial features from the same person lie close in that space.

Formally speaking, in a video segment (150 frames or 5 seconds), we denote each person in that video by an index $n \in [N]$, where N is the total number of people. For each person n, we extract a visual feature g_{VIS} to encode its motion in the video. Meanwhile, we extract a inertial feature g_{IMU} of the target person from the IMU data to encode its motion in 3D space. During training, we learn a visual feature embedding function $H_{\text{VIS}} : g_{\text{VIS}} \rightarrow f$ and a inertial feature embedding function $H_{\text{IMU}} : g_{\text{IMU}} \rightarrow f$ to map both features into the same joint visual-inertial feature space f for matching. At test time, once we find the visual embedding which is the closest to the inertial embedding of a given inertial query in the joint feature space, the target person is localized in the video.

3.1 Visual Feature Extraction

In order to extract people's motion feature from a video segment, we first pre-process the video by performing person detection using YOLOv3 [22] at all frames and then associating the detections into trajectories using a multi-object tracker – DeepSORT

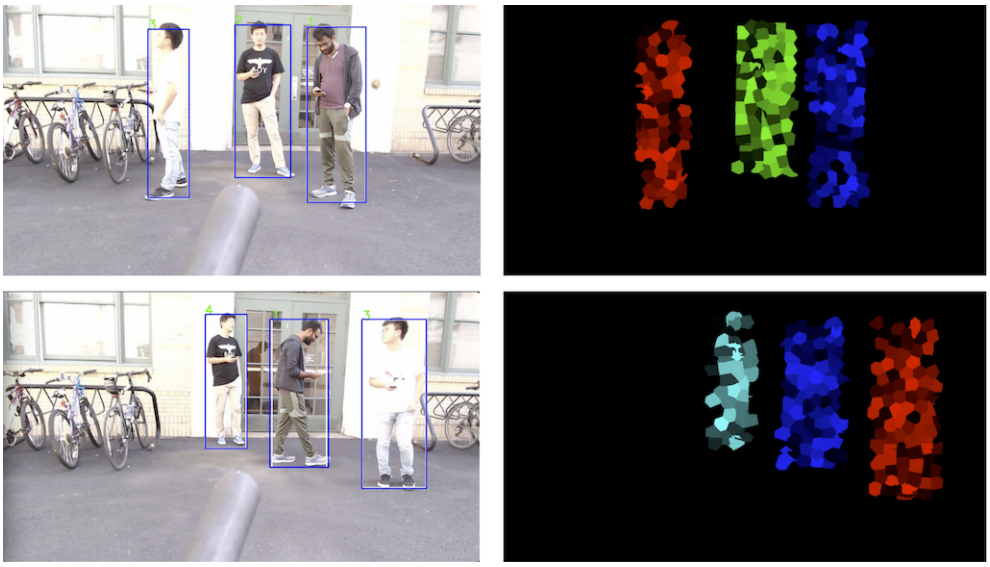

Figure 3.1: **(Left)** YOLOv3 person detections. **(Right)** Temporal super-pixels (TSP) for each tracked person in the video. Average optical flow is computed as the motion feature for each TSP representing different body parts.

[33]. Once we have obtained a trajectory of boxes for each person, we can now extract the motion feature. Specifically, we first extract the optical flow for each box trajectory, and then further decompose it into smaller temporal super-pixels using [5]. The reason for decomposition is that we believe the inertial data measured by the smartphone is only correlated with a part of the body where the smartphone is held, instead of the entire body. Without this decomposition, the optical flow representing the motion of the entire body might not be easily matched with the inertial feature representing the motion of a part of the body, thus leading to inferior localization performance.

Formally speaking, given a video segment $V_{t:t+T}$ with T frames, we denote the set of temporal super-pixels (TSPs) in the video as $\boldsymbol{\xi} = \{\xi_1, \xi_2, \ldots, \}$. We then filter out the TSPs that do not lie within the trajectories of detection boxes and obtain a subset of TSPs denoted as $\boldsymbol{\xi}^n \subset \boldsymbol{\xi}$ for each person. To obtain the motion feature for each TSP, we compute the average optical flow over all pixels for each temporal slice of a TSP:

$$\mathbf{v}_{\xi_i^n} = [(dx_t, dy_t), (dx_{t+1}, dy_{t+1}), ..., (dx_{t+T-1}, dy_{t+T-1})],$$

where each vector $\mathbf{v}_{\xi_i^n}$ represents the motion of a part of the human body as shown

in Fig. 3.1.

Furthermore, for each TSP we compute a sequence of gradients for the average optical flow using finite difference method and the magnitude of gradients at each time step: $|\mathbf{v}_{\xi_i^n}|_t = \sqrt{\dot{dx}_t^2 + \dot{dy}_t^2}$. We use magnitude as an additional feature in order to remove the direction information and only represent the motion intensity in 2D.

Although the TSP features are sufficient to represent the motion information of different body parts in the video, there is still a gap between the TSP features and the inertial 3D motion features as the TSP features are computed in the 2D image space, *i.e.*, the perspective projection of the person's 3D motion. To alleviate this issue and bridge the gap between the 2D and 3D space, we include extra information that is related to the 3D depth and orientation of the person, which can implicitly help the matching between the learned visual and inertial feature embeddings. Specifically, we use two types of information obtained from the video segment:

1. The height and width of the person's bounding box as an indication of the distance to the camera:

$$\mathbf{b}^n = [(h_t, w_t), (h_{t+1}, w_{t+1}), ..., (h_{t+T-1}, w_{t+T-1})].$$

2. The relative positions of the person's left and right shoulder keypoints to the bounding box center as an indication of the body orientation relative to the camera:

$$\mathbf{k}^n = [(\mathbf{ls}_t, \mathbf{rs}_t), (\mathbf{ls}_{t+1}, \mathbf{rs}_{t+1}), (\mathbf{ls}_{t+T-1}, \mathbf{rs}_{t+T-1})],$$

   where **ls** and **rs** are tuples of the keypoint's x and y coordinates relative to the box center's coordinate in the image frame. We choose the shoulder keypoints because their positions are stable to the body orientation.

For the width and height of each person's box, we directly use the box trajectories obtained from YOLOv3 and DeepSORT. To obtain the positions of shoulder keypoints, we first use AlphaPose [34] to detect 17 keypoints of the full body, and then only select the two points representing the shoulder joints. We use linear interpolation to account for occlusion and zero-padding to account for out-of-frame cases.

3.2 Inertial Feature Extraction

To match with the visual motion feature, we also need to extract an inertial feature, which represents the 3D motion of the smartphone for the target person. Given the raw IMU data containing the 3D linear acceleration $\mathbf{a} = [\vec{a}_x, \vec{a}_y, \vec{a}_z]$ and angular velocity $\boldsymbol{\omega} = [\vec{\omega}_x, \vec{\omega}_y, \vec{\omega}_z]$ in the smartphone's local coordinate frame, we construct the inertial feature for target person n denoted as $g^n_{\text{IMU}} = [\vec{a}_x, \vec{a}_y, \vec{a}_z, \vec{\omega}_x, \vec{\omega}_y, \vec{\omega}_z]^T$ by concatenating linear acceleration and angular velocity. As a result, the inertial feature g^n_{IMU} is a $6 \times M$ matrix where M is the number of frames temporally aligned with the video segment's time window. As the IMU frame rate is 100Hz, with a ratio of 3.33:1 to the video frame rate of 30Hz, we uniformly sample the inertial frames so that $M = 3 \times T$, where T is the number of frames in a video segment. Furthermore, we apply a low-pass filter to reduce high frequency noise in the raw IMU data. Similar to computing the magnitude of optical flow gradient in visual feature extraction, we also compute the magnitude of linear accelerations to represent 3D motion intensity $|\mathbf{a}|_t = \sqrt{a_x^2 + a_y^2 + a_z^2}$.

3.3 Learning the Visual-Inertial Feature Space

Although the raw visual feature and the raw inertial feature contain sufficient information representing the person's 3D motion, they still lie in different feature spaces as they are obtained from different source of data and thus it is difficult to directly match them. To overcome this issue, we propose to learn a feature transformer that further transform the raw visual and inertial features into a joint feature space so that the matching for a same person is possible.

The proposed network for learning the joint feature space is shown in Fig. 3.2. To transform the raw visual feature into the joint space while model the temporal dependency, we first apply four LSTM networks for the TSP optical flow features (LSTM-OpticalFlow), bounding box size data (LSTM-Box), pose keypoints data (LSTM-Pose), and optical flow magnitude (LSTM-OF-Mag). Then, we concatenate hidden state output from LSTM-OpticalFlow, LSTM-Box and LSTM-Pose, and feed it through a fully-connected layer to produce the final output embedding. We keep the magnitude feature embedding from LSTM-OF-Mag as a separate branch for

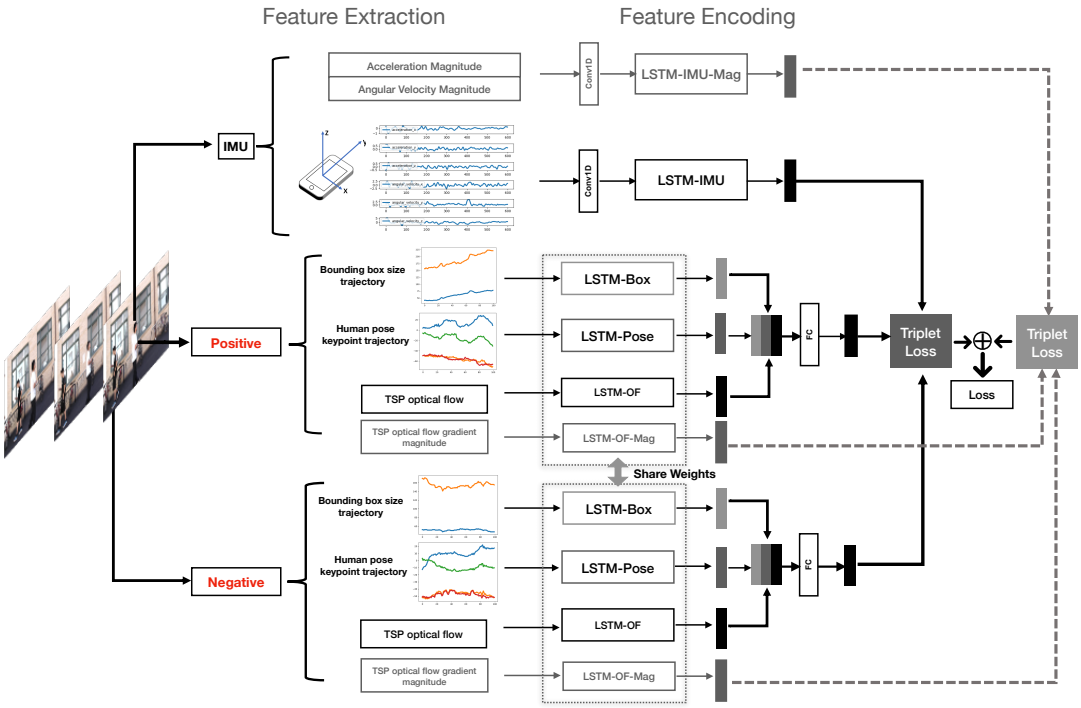

Figure 3.2: **Proposed Network**. Our network has one branch to extract the inertial feature of the target person and two branches to extract the visual features from one positive and one negative sample. At each iteration of training, the positive visual feature is extracted from the target person while the negative visual feature is from a randomly picked different person. Once the raw inertial and visual features are extracted, they are fed into our visual-inertial feature transformer so that the transformed feature embeddings lie in a same feature space. A triplet loss is then applied to minimize the L2 distance between the inertial embedding and the positive visual embedding while maximize the L2 distance between the inertial embedding and the negative visual embedding. At test time, we compute the visual embeddings for all persons in the video and also compute the inertial embedding of the target person. The predicted target person in the video is then the person whose visual embedding has minimum distance to the target person's inertial embedding.

matching with inertial magnitude embedding directly.

To transform the raw inertial feature into the joint space, we first use a 1D convolution layer to reduce the dimensionalities of the inertial feature to be the same as the visual feature. Then, we also apply an LSTM network (LSTM-IMU) to model the temporal dependency for the inertial feature. Similarly, we apply a 1D convolution layer and an LSTM (LSTM-IMU-Mag) to encode the inertial magnitude feature.

Formally, the visual and inertial (magnitude) feature is defined as:

$$H_{\text{VIS}}(\mathbf{v}_{\xi_i^n}, \mathbf{b}^n, \mathbf{k}^n) = f_{\text{VIS}}[f_{\text{OF}}(\mathbf{v}_{\xi_i^n}) \oplus f_{\text{Pose}}(\mathbf{k}^n) \oplus f_{\text{Box}}(\mathbf{b}^n)],$$

$$H_{\text{VIS-mag}}(|\mathbf{v}_{\xi_i^n}|) = f_{\text{VIS-mag}}(|\mathbf{v}_{\xi_i^n}|),$$

$$H_{\text{IMU}}(g_{\text{IMU}}^n) = f_{\text{IMU}}(g_{\text{IMU}}^n),$$

$$H_{\text{IMU-mag}}(|\mathbf{a}|) = f_{\text{IMU-mag}}(|\mathbf{a}|),$$

For training, since each person n has a set of TSPs $\boldsymbol{\xi}^n$ and thus we have $|\boldsymbol{\xi}^n|$ final visual embeddings, we duplicate the number of final inertial embeddings so that we have the same number of visual and inertial embeddings for each person in a time window with T frames. We use every pair of the inertial and visual embeddings and minimize the L2 distance between them if they belong to the same person.

Furthermore, we use the triplet loss as in [10, 12]. Specifically, for each target person n with the inertial embedding, we use the visual embedding obtained from the same target person as a positive example and use the visual embedding obtained from a randomly sampled different person as a negative example. The positive and negative samples share the same weights in the LSTM networks (*i.e.*, LSTM-OF, LSTM-Pose, LSTM-Box, LSTM-OF-Mag). Then, the triplet loss is applied to minimize the L2 distance between the inertial and positive visual embedding and maximize the L2 distance between the inertial and negative visual embedding:

$$\begin{aligned}\mathcal{L}_1(g_{\text{IMU}}^n, g_{\text{VIS}}^+(\xi_i), g_{\text{VIS}}^-(\xi_j)) = \max(&||H_{\text{VIS}}(g_{\text{VIS}}^+(\xi_i)) - H_{\text{IMU}}(g_{\text{IMU}}^n)||_2 - \\ &||H_{\text{VIS}}(g_{\text{VIS}}^-(\xi_j)) - H_{\text{IMU}}(g_{\text{IMU}}^n)||_2 + \kappa_1, 0).\end{aligned}$$

We use a separate triplet loss for learning the visual and inertial magnitude embeddings as they represent only the intensity of the motion:

$$\mathcal{L}_2(|\mathbf{a}^n|, |\mathbf{v}^+_{\xi_i^n}|, |\mathbf{v}^-_{\xi_i^n}|) = \max(||H_{\text{VIS-mag}}(|\mathbf{v}^+_{\xi_i^n}|) - H_{\text{IMU-mag}}(|\mathbf{a}^n|)||_2 - ||H_{\text{VIS-mag}}(|\mathbf{v}^-_{\xi_i^n}|) - H_{\text{IMU-mag}}(|\mathbf{a}^n|)||_2 + \kappa_2, 0),$$

κ_1, κ_2 are the margins separating the positive and negative feature space. The final loss is defined to be weighted sum over the two triplet losses:

$$\mathcal{L} = \alpha\mathcal{L}_1 + (1-\alpha)\mathcal{L}_2.$$

In our experiments, we use $\alpha = 0.5$.

At test time, given a video segment $V_{t:t+T}$ with N people in the scene, we choose one person as the target person at a time and compute its inertial embedding. Meanwhile, we compute the visual embedding for all persons in the video. Then, the predicted target person is the person whose visual embedding averaged over all TSPs has the minimum distance to the target person's inertial embedding:

$$\hat{n} = \arg\min_{n' \in [N]} \frac{1}{|\boldsymbol{\xi}^{n'}|} \sum_{i=1}^{|\boldsymbol{\xi}^{n'}|} \alpha||H_{\text{VIS}}(g^{n'}_{\text{VIS}}(\xi_i)) - H_{\text{IMU}}(g^n_{\text{IMU}})||_2 + (1-\alpha)||H_{\text{VIS-mag}}(|\mathbf{v}_{\xi_i}|) - H_{\text{IMU-mag}}(|\mathbf{a}^n|)||_2,$$

where $|\boldsymbol{\xi}^{n'}|$ is the number of TSP's for person n'.

Chapter 4

Dataset

To train our proposed method for visual-inertial person localization in the wild, we need a dataset with synchronized video and inertial data that include multiple people acting freely outside, each carrying a smartphone in their hand. However, existing visual-inertial datasets [8, 9, 24] do not satisfy these requirements and often have three limitations: 1) they rigidly attach the inertial sensor to person's body (*e.g.*, limb or back) so that the motion of the inertial sensor tightly aligns with the body part; 2) they often record the data in the indoor setting; 3) only one person is recorded at one time. As a result, prior datasets are not applicable to our challenging visual-inertial person localization task, and we collected a new dataset to satisfy the task conditions.

4.1 Video Recording

We set up a HD webcam with a resolution of 1920×1080 on a tripod about one meter above the ground for video recording, similar to the setting of a dashboard camera in a car. We choose to record the video outside public buildings in order to mimic the real world autonomous vehicle pickup scenarios. At each time of the recording, we

Table 4.1: Statistics of the video data collected in our dataset.

Number of people	2	3	4	5	6
Number of videos	17	15	11	7	8
Number of total frames	12,900	19,600	21,400	10,084	5,000

hire 2-6 different volunteers and assign a smartphone to each of the volunteer during the video recording. Each video recording is about half to two minutes long with a frame rate of 30Hz. In total, we have recorded 58 videos with a total of 68984 frames. We summarize the statistics of our data recording in Table 4.1. Our dataset contains common types of pedestrian motion such as standing, walking and turning, recorded in front of different buildings to increase the diversity of the dataset. As we record the data in the wild, we also allow random people to appear in the video without recording their inertial data in order to mimic the challenging real-world scenario. Also, we do not provide and allow to use the calibration parameters of the camera in our dataset, as in the real world the calibration parameters of the dashboard camera might vary across vehicles and not available to our approach for person localization. Each video frame is time-stamped with the UTC time for synchronization with IMU.

4.2 IMU Recording

We use iPhone (model 7 and 8) as the smartphone device to collect the inertial data. To that end, we have developed an iOS application with the iOS Core Motion Framework to obtain the linear acceleration and angular velocity data from the onboard accelerometer and gyroscope. For linear acceleration, we use the processed data by the device that only reflects the user-generated acceleration after removing the gravity. The IMU data is recorded at 100Hz with UTC timestamps. At each time of the recording, we ask the volunteers to start the iOS application on their iPhones so that the data can be saved to the device. As the data synchronization is handled by matching the timestamp, volunteers do not need to start the application exactly at the same time.

4.3 Data pre-processing

As optical flow is needed to obtain the visual embedding, we pre-compute the flow for all videos in advance so that the online training can be faster. However, computing optical flow on the raw images with a resolution of 1920×1080 is very expensive, we thus downsample the raw images to a resolution of 691×389 to speed up the pre-

processing step. Also, as our network can only process a short video segment at a time, we convert the raw video and inertial data into short segments using a sliding window approach. Specifically, we experiment with a window size of $\{100, 150, 180, 200\}$ and step size of 20 frames. As a result, over $60,000$ synchronized video and inertial data segments are generated.

As we have the inertial data for all persons in each data segment, we can iteratively mark each person in the data segment as the target person. This means that each video segment can serve as m data segment samples during training and evaluation where m equals to the number of persons in the video. This data augmentation technique further increases the number of our data segment samples about four times.

Chapter 5

Experiments

5.1 Evaluation Details

Since our visual-inertial person localization is formalized as a matching problem, we use the classification rate as our evaluation metric, namely the probability that our method can output a correct match for the target person. We split our collected data into train, validation and test set, where each set contains videos with different number of people. The evaluation of our method and baselines is only conducted on the test set, while the validation set is used for parameter tuning. Usually, when there are more people in the scene, it is more likely that people will have similar motion (*e.g.*, walking in the same direction), which makes the data more difficult for matching and localization. Naturally, we expect that our visual-inertial person localization task to become harder when there are more people in the scene. We show quantitative results with $\{2, 3, 4, 5\}$ people in the scene.

5.2 Comparison to Baseline Methods

Since this is the first work to address the task of visual-inertial source localization, we are not able to compare against any available state-of-the-art method. Instead we have designed several features both hand-designed and learned to compare with our method. In each of our experiments we use a temporal window of size 5 seconds

which is $K = 150$ video frames.

5.2.1 Non-learning Visual-Inertial Matching

We perform direct matching between the processed visual and inertial data sequence within the aligned temporal window. Specifically, given the query IMU sequence, we compute the cosine distance between the inertial sequence and all visual sequences that belong to the candidate people in the video, and predict the target IMU source person with the minimum feature distance. We design the following four processing approaches to generate the feature sequences:

Velocity Magnitude Sequence: For the visual feature f_{vis}, we compute a sequence of magnitudes of the optical flow for each TSP, $\{(v_x^2+v_y^2)^{1/2}\}_{k=1}^{K}$. Likewise for the IMU feature f_{imu}, we first compute the 3D velocity from 3D linear acceleration measured by the IMU using integration, $\vec{v}_t = \vec{v}_{t-1} + \vec{a}_t \nabla t$. Then we compute a sequence of velocity magnitudes $\{||\vec{v}_t||_2\}_{k=1}^{K}$ for the IMU signal. The IMU signal is down-sampled to ensure that the dimensions match with the visual feature.

Acceleration Magnitude Sequence: For the visual feature f_{vis}, we compute a sequence of gradient magnitudes of the optical flow for each TSP, $\{(a_x^2 + a_y^2)^{1/2}\}_{k=1}^{K}$. Likewise for the IMU feature f_{imu}, we compute a sequence of linear acceleration magnitudes $\{||\vec{a}_t||_2\}_{k=1}^{K}$. The IMU signal is down-sampled to ensure that the dimensions match with the visual feature.

Velocity Magnitude Histogram: The computation of the velocity magnitudes follows the velocity magnitude sequence features described above but velocity magnitudes are binned to create a velocity magnitude histogram, where the range of velocities has been equally divided into 100 bins.

Acceleration Magnitude Histogram: The computation of the acceleration magnitudes follows the magnitude sequence features described above but acceleration magnitudes are binned to create a magnitude histogram, where the range of accelerations has been equally divided into 100 bins.

5.2.2 Supervised-Learning for Single Modality

3D Orientation Sequence: We re-implemented image-based orientation estimation technique in [9] where the person's 3D orientation is predicted from a VGG16-extended

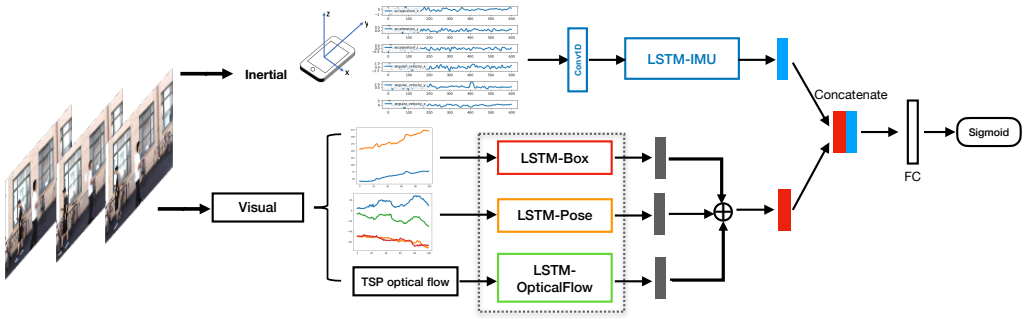

Figure 5.1: Visual-Inertial Binary Classification

network with RGB image input. We designed a similar network based on VGG16 by adding two fully connected layers to learn the mapping from image to the person orientation. We used a concatenated person image pair in the tracklet as input and trained the network to regress the temporally-aligned orientation change using angular velocity measurement from IMU as ground truth. This results in a sequence of 3D orientations as the visual features $f_{\text{vis}} = \{\vec{v}_t\}_{t=1}^{T}$. The IMU signal is down-sampled to ensure that the dimensions match with the visual feature.

Optical Flow Sequence: For the visual feature f_{vis}, we compute a sequence of magnitudes of the optical flow for each TSP, $\{(v_x^2 + v_y^2)^{1/2}\}_{k=1}^{K}$. The IMU feature f_{imu} is extracted by mapping a sequence of acceleration $\vec{a}$ and angular velocity $\vec{w}$ to a sequence of 2D velocities. The mapping function is learned by supervised learning where the observation is $(\vec{a}, \vec{\omega})$ and the label is the optical flow. The IMU feature is a sequence of optical flow estimated from the IMU measurements.

5.2.3 Visual-Inertial Binary Classification

We formulate a binary classification problem for predicting a single probability of whether the given visual-inertial features match. We extract the same features from IMU and video and learn the embeddings as described in our proposed method. Instead of formulating a triplet with positive and negative samples during training, we only concatenate the inertial and visual features that belong to the same person, and feed it through a fully-connected layer and a sigmoid function to output a single score. During testing, given a query IMU, we generate scores between the inertial feature and all visual features from all persons in the video, and take the person with

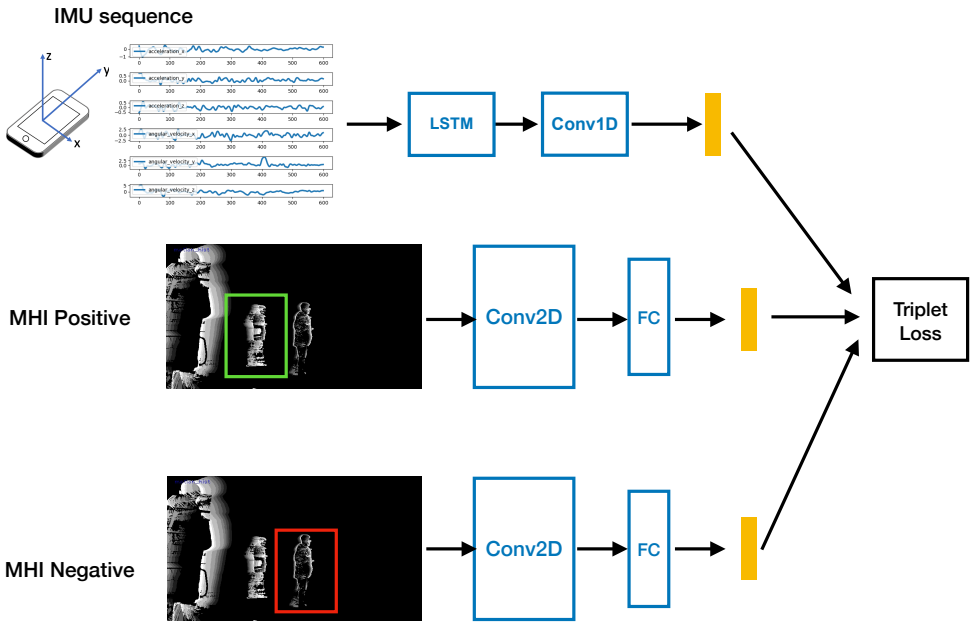

Figure 5.2: Triplet Model with Motion History Image as Visual Representation

highest probability as the predicted IMU source.

5.2.4 Triplet Model with Motion History Image as Visual Representation

Motion History Image (MHI) [2] is a compact way of representing temporal motion information with a static image, where the pixel intensity is a function of recency of motion. It is widely used for action recognition and motion analysis [1, 7, 25]. MHI is designed to be sensitive to temporal motion at pixel level while preserving dominant motion information. We use MHI in place of the visual features with the triplet model as in Figure 5.2, as another baseline to explore a different visual representation of human motion. Specifically, for each frame in the video segment of size 150, we generate an MHI with a history of 20 frames and the synchronized IMU as the input to our model, with the objective of minimizing the triplet loss. At test time, we compute the feature distance between all MHIs and IMU sequences in the temporal window and predict the IMU source with the minimum average feature distance.

Table 5.1: Quantitative comparison of our method with baselines.

Number of People	N=2	N=3	N=4	N=5	Overall
Number of Samples	32	135	102	163	432
Random Guess	0.500	0.333	0.250	0.200	0.276
1) Velocity Magnitude	0.474	0.333	0.294	0.261	0.307
2) Velocity Mag. Histogram	0.519	0.333	0.426	0.291	0.352
3) Acceleration Magnitude	0.766	0.495	0.738	0.313	0.504
4) Accel. Mag. Histogram	0.519	0.366	0.455	0.321	0.381
5) 3D Orientation [9]	0.502	0.344	0.306	0.194	0.290
6) 2D Optical Flow	0.682	0.402	0.392	0.439	0.434
7) Visual-Inertial Binary Classification	0.667	**0.697**	0.635	0.605	0.645
8) MHI-Inertial Triplet	**0.950**	0.667	0.717	**0.692**	0.709
Ours	0.875	0.681	**0.784**	0.688	**0.724**

We show quantitative comparison of our method and above baselines in Table 5.1. We can see that baseline methods 1 to 4 with hand-designed feature often perform poorly as the motion features from the visual and inertial modalities are in different feature spaces, and it is challenging to directly match them. Also, learning to transform one modality to the other (*i.e.*, baseline methods 5 and 6) does not achieve superior performance. This proves again the significant gap between the two modalities. We show that, only when we transform the features from two modalities into a joint feature space in our method, significant improvement can be achieved across videos with different number of people in the scene. Binary classification trained with only visual-inertial data from the same person achieves competitive results but still worse than our method, which shows effectiveness of using a triplet loss to reduce the number of false positives. MHI-Inertial Triplet model has high accuracy on some cases but the overall performance is slightly worse than ours, which means MHI is not a consistently good visual feature representation. Practically, MHI would suffer more from camera ego-motion if the system is extended for videos captured from a vehicle dash camera, which makes it a less-than-ideal design choice for visual feature representation. Additionally, we show the qualitative results of our method on the test set in Fig. 5.3. The results show that our method can predict a correct match in most of the frames, while in the failure cases the true target is

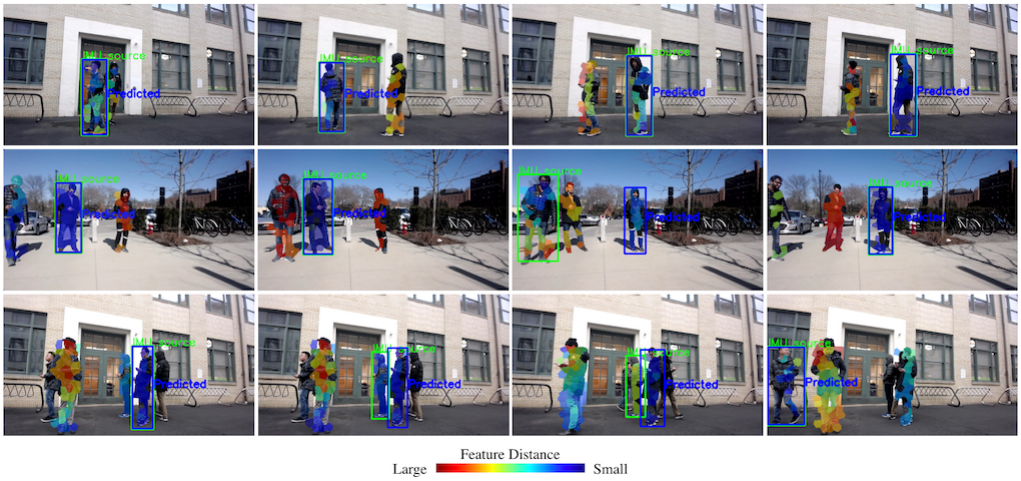

Figure 5.3: We show qualitative results of our method for visual-inertial person localization on three test videos with different number of people in the scene. The green box indicated as the IMU source is the target person while the blue box is the predicted target person by our method. When the green and blue boxes fall on a same person, it is a correct match. We show both successful and failure cases in the results. Also, we visualize the distance of the visual feature for each TSP to the inertial feature of the true target person.

often confused with another false predicted target with similar motion (best viewed in video).

5.3 Ablation Study

5.3.1 Length of the Time Window

As more discriminative motion feature can be found in a longer time window, we believe the length of time window is an important factor to the performance of our method and run ablation experiments with respect to it. Specifically, we run experiments with a window length of $100, 150, 180, 200$ frames (*i.e.*, $3, 5, 6, 6.67$ seconds). We use the same step size of 20 frames (0.67 seconds) for all experiments. We found that the highest accuracy is achieved with a window length of 150 frames. Also, we observed a performance drop when the window length goes beyond 150 frames. It turns out that when the window length increases beyond 150 frames, the number of data samples

Table 5.2: Performance of our method with respect to window length.

Window Length / frames	Number of Training Samples	Overall Accuracy
100	513	0.622
150 (**Ours**)	432	**0.724**
180	397	0.427
200	363	0.568

Table 5.3: Performance of our method with respect to different variations of the inertial feature representation.

Inertial Feature Representation	N=2	N=3	N=4	N=5	Overall
$(\hat{\mathbf{v}}, \mathbf{a}, \boldsymbol{\omega})$	0.719	0.563	0.637	0.656	0.627
$(\mathbf{a}, \boldsymbol{\omega})$	0.750	0.632	0.643	0.619	0.638
$(\hat{\mathbf{v}}, \boldsymbol{\omega})$	0.656	0.495	0.461	0.405	0.465
$(\mathbf{a}_{\mathrm{LPF}}, \boldsymbol{\omega}_{\mathrm{LPF}})$ (**Ours**)	**0.875**	**0.681**	**0.784**	**0.688**	**0.724**

drops significantly as most of the person trajectories in our dataset are short due to heavy occlusion by other persons. As a result, due to limited data samples, training process of our network becomes unstable and evaluation is not trustable. Additionally, a longer time window means a larger latency of our method. Therefore, we did not further investigate longer time window but use the window of 150 frames in our final model.

5.3.2 Inertial Feature Representation

The use of a different feature representation can result in significant differences in performance. Here, we first investigate different variations of the inertial feature representation. In addition to the linear acceleration and angular velocity, we believe the linear velocity might be also an informative feature for matching with the visual motion feature. To that end, we integrate the linear acceleration from the IMU to estimate the linear velocity $\hat{\mathbf{v}} = [\hat{v}_x, \hat{v}_y, \hat{v}_z]$ as an additional 3D motion information. As we ask the volunteers to stand still at the beginning of each video recording and then to start moving freely, we can use an initial velocity of 0 for the integration. Results in Table 5.3 first row ($\hat{\mathbf{v}}$, $\mathbf{a}$, $\boldsymbol{\omega}$) show that concatenating the estimated linear velocity with the linear acceleration and angular velocity unfortunately performs slightly worse than without adding the linear velocity as shown in the second row

Table 5.4: Performance of our method with respect to different variations of the visual feature representation.

Visual Feature Representation	N=2	N=3	N=4	N=5	Overall
TSP optical flow w/ **b** and **k** (**Ours**)	**0.875**	**0.681**	**0.784**	**0.688**	**0.724**
TSP optical flow w/ **b**	0.850	0.562	0.529	0.380	0.507
TSP optical flow w/ **k**	0.719	0.632	0.637	0.405	0.554
TSP optical flow only	0.624	0.504	0.536	0.312	0.448

Table 5.5: Performance of our method with/without visual-inertial magnitude features.

Visual-Inertial Magnitude	N=2	N=3	N=4	N=5	Overall
With (**Ours**)	**0.875**	**0.681**	**0.784**	0.688	**0.724**
Without	0.767	0.671	0.641	**0.720**	0.689

of Table 5.3. Also, we experiment a variant that concatenates the estimated linear velocity and angular velocity in the third row of Table 5.3, which has a even lower performance than both the first and second row. These results demonstrate that the estimated linear velocity through integration might not be accurate enough due to the error accumulation from the IMU drift and thus we do not use the linear velocity in our final model.

Additionally, as the inertial data obtained from the IMU sensor often has high-frequency noise, we experiment the effect of a low-pass filter to our method. Specifically, we apply the filter to both the linear acceleration and angular velocity and obtain a smoother version of the inertial features $(\mathbf{a}_{\text{LPF}}, \boldsymbol{\omega}_{\text{LPF}})$, which turns out improving overall performance by 13.5% across settings with different number of people.

5.3.3 Visual Feature Representation

To verify whether adding the relative positions of person's shoulder keypoints and the bounding box size to the visual feature is useful in our model, we ran experiments with both features, either feature, or none in addition to the optical flow feature of the temporal super-pixels. From the results in Table 5.4, we observed that both shoulder keypoints and bounding box size features are indeed useful and improve the performance on test videos with different number of people.

Table 5.6: Performance with regard to different levels of motion.

Number of People	N=2	N=3	N=4	N=5	Overall
Number of samples	5	45	34	29	113
$\gamma \leq 0.4$	1.000	0.333	0.588	0.275	0.424
Number of samples	27	90	68	134	319
$\gamma > 0.4$	0.815	0.867	0.662	0.813	0.796

5.3.4 Visual-Inertial Magnitude Feature

We also compare models with and without the magnitude features from optical flow and IMU acceleration as a separate triplet branch, to verify whether it is necessary to include the motion intensity information in our model. From results in Table 5.5, adding magnitude features improves overall performance by 5%.

5.3.5 Performance with regard to Different Levels of Motion

From the quantitative results for baseline comparison and ablation studies, we observe our presumption that larger number of people would cause decrease in prediction accuracy does not always hold. As our proposed method relies on the motion feature matching between the inertial and visual modalities, it is difficult to perform the matching if the target person has nearly no motion. Therefore, we introduce a new task complexity measurement — motion diversity. We define the threshold as the sum of standard deviations of IMU acceleration and angular velocity sequences over the time window: $\sigma(\{||\boldsymbol{\omega}||_2\}_{k=1}^{K}) + \sigma(\{||\mathbf{a}||_2\}_{k=1}^{K}) < \gamma$.

We apply a threshold of $\gamma = 0.4$ to filter testing samples with large and small variations in acceleration and angular velocity. Table 5.6 shows performance of our proposed method on testing set separated by large and small motions. The prediction accuracy is significantly higher where $\gamma > 0.4$, and fails to achieve over 50% for $\gamma \leq 0.4$. In the future, we plan to deal with this limitation with additional small-motion-sensitive or context features in order to achieve person localization even the target person has no motion.

Chapter 6

Conclusions

6.1 Summary

We explore the possibility of using the inertial data to localize the target person in the video, in the case where we do not have access to the target person's appearance information in advance. We term this proposed task as the visual-inertial person localization. To solve this task, we first collect a new large visual-inertial dataset, which is significantly different from existing datasets in that our new dataset contains multiple people in the wild and does not have strict constraint on the attached location of the inertial sensor. Additionally, we propose an effective approach that learns a transformer and maps the visual and inertial features into a joint feature space for matching. Through extensive experiments, we show effectiveness of each component of our method and demonstrate that the proposed method outperforms competitive baselines in our challenging dataset.

6.2 Future work

1. Currently, although the dataset contains certain amount of natural and diverse pedestrian motions, it is still limited in size in terms of number of people and scenes. Collecting more data at scale is an important next step to see how the model generalizes to more complex scenarios (*e.g.*more occlusions where there

can be more uncertainty in person detection and tracking), and it is necessary for training the model to avoid overfitting.

2. As our final goal is to deploy this system to mobile robots or autonomous vehicles for locating specific people in the video stream given their IMUs, one potential extension to this work is to test on video captured from a moving camera. Currently each video segment in the dataset used for training and testing is captured from a webcam fixed on a tripod. For future data collection, video can be captured from a car-mounted camera to simulate the car approaching a pick-up location. To adapt our current framework to dynamic videos, further modification to the model might be necessary, such as background segmentation and ego-motion estimation, to remove noise from ego-motion.

Bibliography

[1] Md Atiqur Rahman Ahad. *Motion history images for action recognition and understanding*. Springer Science & Business Media, 2012. 5.2.4

[2] Md Atiqur Rahman Ahad, Joo Kooi Tan, Hyoungseop Kim, and Seiji Ishikawa. Motion history image: its variants and applications. *Machine Vision and Applications*, 23(2):255–281, 2012. 5.2.4

[3] Claudine Badue, Rânik Guidolini, Raphael Vivacqua Carneiro, Pedro Azevedo, Vinicius Brito Cardoso, Avelino Forechi, Luan Jesus, Rodrigo Berriel, Thiago Paixão, Filipe Mutz, Lucas Veronese, Thiago Oliveira-Santos, and Alberto Ferreira De Souza. Self-Driving Cars: A Survey. *arXiv:1901.04407*, 2019. 1

[4] Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, and Ben Upcroft. Simple Online and Realtime Tracking. *ICIP*, 2016. 1

[5] Jason Chang, Donglai Wei, and John W. Fisher, III. A Video Representation Using Temporal Superpixels. *CVPR*, 2013. 1, 3.1

[6] Tianlong Chen, Shaojin Ding, Jingyi Xie, Ye Yuan, Wuyang Chen, Yang Yang, Zhou Ren, and Zhangyang Wang. ABD-Net: Attentive but Diverse Person Re-Identification. *ICCV*, 2019. 2.3

[7] James W Davis. Hierarchical motion history images for recognizing human motion. pages 39–46, 2001. 5.2.4

[8] Fernando de la Torre, Jessica K. Hodgins, Javier Montano, and Sergio Valcarcel. Detailed Human Data Acquisition of Kitchen Activities: the CMU-Multimodal Activity Database (CMU-MMAC). *CHI Workshop*, 2009. 2.2, 4

[9] Roberto Henschel, Timo von Marcard, and Bodo Rosenhahn. Simultaneous Identification and Tracking of Multiple People Using Video and IMUs. *CVPRW*, 2019. 1, 2.1, 4, 5.2.2, 5.1

[10] Alexander Hermans, Lucas Beyer, and Bastian Leibe. In Defense of the Triplet Loss for Person Re-Identification. *arXiv:1703.07737*, 2017. 3.3

[11] Sepp Hochreiter and Jj Urgen Schmidhuber. Long Short-Term Memory. *Neural Computation*, 1997. 1

[12] Elad Hoffer and Nir Ailon. Deep Metric Learning Using Triplet Network. *International Workshop on Similarity-Based Pattern Recognition*, 2015. 3.3

[13] Yinghao Huang, Manuel Kaufmann, Emre Aksan, Michael J. Black, Otmar Hilliges, and Gerard Pons-Moll. Deep Inertial Poser: Learning to Reconstruct Human Pose from Sparse Inertial Measurements in Real Time. *SIGGRAPH Asia*, 2018. 1

[14] Seita Kayukawa, Keita Higuchi, Joao Guerreiro, Shigeo Morishima, Yoichi Sato, Kris Kitani, and Chieko Asakawa. BBeep: A Sonic Collision Avoidance System for Blind Travellers and Nearby Pedestrians. *CHI*, 2019. 1

[15] B Ravi Kiran, Ibrahim Sobh, Victor Talpaert, Patrick Mannion, Ahmad A. Al Sallab, Senthil Yogamani, and Patrick Pérez. Deep Reinforcement Learning for Autonomous Driving: A Survey. *arXiv:2002.00444*, 2020. 1

[16] Florin Leon and Marius Gavrilescu. A Review of Tracking, Prediction and Decision Making Methods for Autonomous Driving. *arXiv:1909.07707*, 2019. 1

[17] Jianing Li, Jingdong Wang, Qi Tian, Wen Gao, and Shiliang Zhang. Global-Local Temporal Representations For Video Person Re-Identification. *ICCV*, 2019. 2.3

[18] Yu-Jhe Li, Zhengyi Luo, Xinshuo Weng, and Kris M. Kitani. Learning Shape Representations for Clothing Variations in Person Re-Identification. *arXiv:2003.07340*, 2020. 2.3

[19] Wenjie Luo, Bin Yang, and Raquel Urtasun. Fast and Furious: Real Time End-to-End 3D Detection, Tracking and Motion Forecasting with a Single Convolutional Net. *CVPR*, 2018. 1

[20] Aashi Manglik, Xinshuo Weng, Eshed Ohn-bar, and Kris M Kitani. Forecasting Time-to-Collision from Monocular Video: Feasibility, Dataset, and Challenges. *IROS*, 2019. 1

[21] Akshay Rangesh and Mohan M. Trivedi. No Blind Spots: Full-Surround Multi-Object Tracking for Autonomous Vehicles using Cameras & LiDARs. *IV*, 2019. 1

[22] Joseph Redmon and Ali Farhadi. YOLOv3: An Incremental Improvement. *arXiv:1804.02767*, 2018. 3.1

[23] Florian Schroff, Dmitry Kalenichenko, and James Philbin. FaceNet: A Unified Embedding for Face Recognition and Clustering. *CVPR*, 2015. 1

[24] Matt Trumble, Andrew Gilbert, Charles Malleson, Adrian Hilton, and John Collomosse. Total Capture: 3D Human Pose Estimation Fusing Video and Inertial Sensors. *BMVC*, 2017. 2.2, 4

[25] Michel Valstar, Maja Pantic, and Ioannis Patras. Motion history for facial action detection in video. 1:635–640, 2004. 5.2.4

[26] Timo von Marcard, Bodo Rosenhahn, Michael Black, and Gerard Pons-Moll. Sparse Inertial Poser: Automatic 3D Human Pose Estimation from Sparse IMUs. *Eurographics*, 2017. 1

[27] Timo von Marcard, Roberto Henschel, Michael J. Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering Accurate 3D Human Pose in the Wild Using IMUs and a Moving Camera. *ECCV*, 2018. 1, 2.4

[28] Sen Wang, Daoyuan Jia, and Xinshuo Weng. Deep Reinforcement Learning for Autonomous Driving. *arXiv:1811.11329*, 2018. 1

[29] Xinshuo Weng and Kris Kitani. A Baseline for 3D Multi-Object Tracking. *arXiv:1907.03961*, 2019. 1

[30] Xinshuo Weng, Jianren Wang, Sergey Levine, Kris Kitani, and Nicholas Rhinehart. Unsupervised Sequence Forecasting of 100,000 Points for Unsupervised Trajectory Forecasting. *arXiv:2003.08376*, 2020. 1

[31] Xinshuo Weng, Yongxin Wang, Yunze Man, and Kris Kitani. GNN3DMOT: Graph Neural Network for 3D Multi-Object Tracking with 2D-3D Multi-Feature Learning. *CVPR*, 2020. 1

[32] Xinshuo Weng, Ye Yuan, and Kris Kitani. Joint 3D Tracking and Forecasting with Graph Neural Network and Diversity Sampling. *arXiv:2003.07847*, 2020. 1

[33] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple Online and Realtime Tracking with a Deep Association Metric. *ICIP*, 2017. 1, 3.1

[34] Yuliang Xiu, Jiefeng Li, Haoyu Wang, Yinghong Fang, and Cewu Lu. Pose Flow: Efficient Online Pose Tracking. *BMVC*, 2018. 3.1

[35] Ekim Yurtsever, Jacob Lambert, Alexander Carballo, and Kazuya Takeda. A Survey of Autonomous Driving: Common Practices and Emerging Technologies. *arXiv:1906.05113*, 2019. 1

[36] Wenyuan Zeng, Wenjie Luo, Simon Suo, Abbas Sadat, Bin Yang, Sergio Casas, and Raquel Urtasun. End-to-End Interpretable Neural Motion Planner. *CVPR*, 2019. 1

[37] Kaiyang Zhou, Yongxin Yang, Andrea Cavallaro, and Tao Xiang. Omni-Scale Feature Learning for Person Re-Identification. *ICCV*, 2019. 2.3