1. Multi-modal Pedestrian Dataset

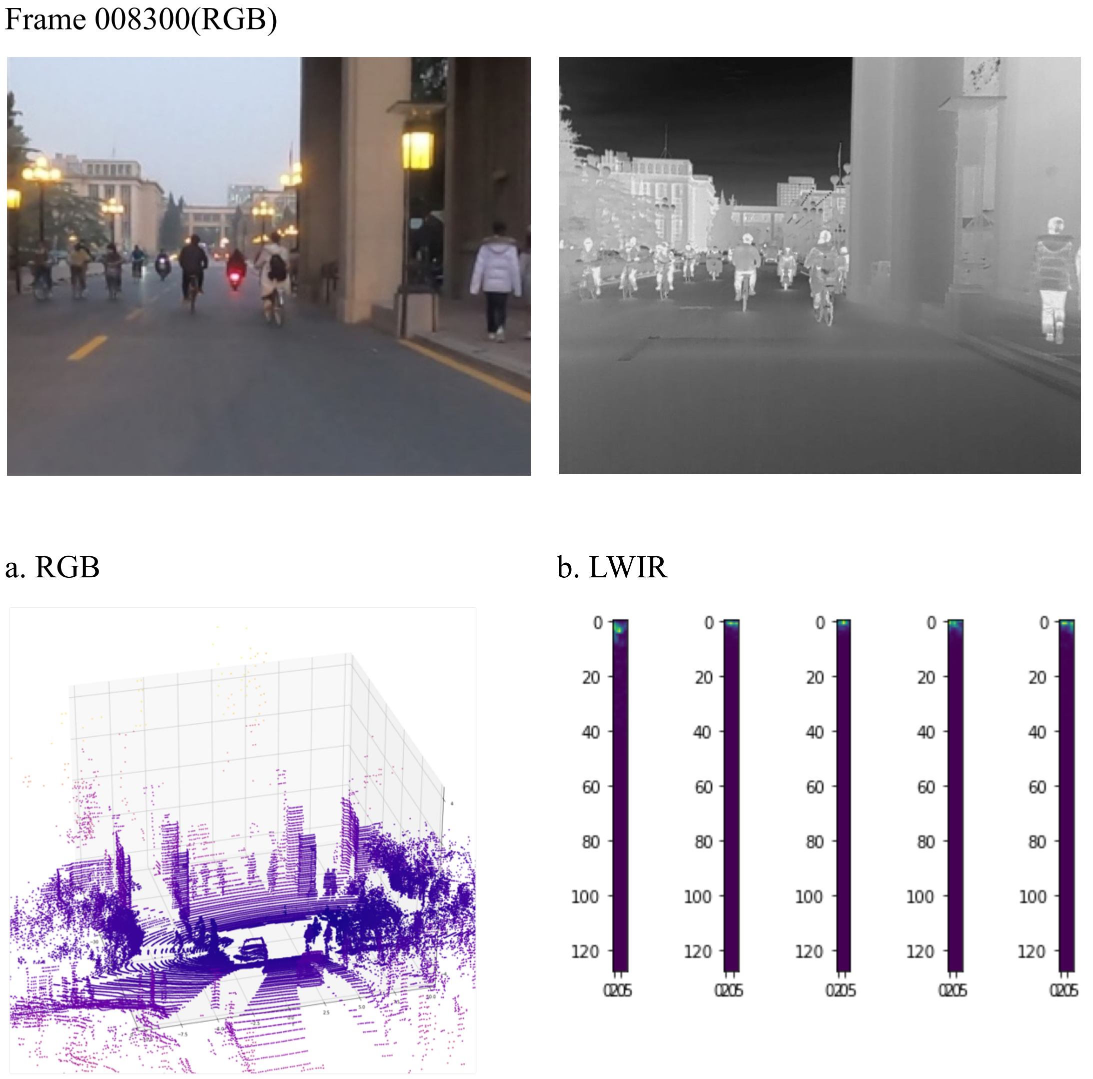
Currently There are a lot of existing datasets with collected with different vehicles, in different cities and with different sensory input**Error! Reference source not found.Error! Reference source not found.Error! Reference source not found.**. Yet, the majority of which only consist limited modality and cannot accurately portrait the road condition of china. Also, most datasets are constructed with autonomous driving in mind, the data have an innate bias toward multi-vehicle environments, which are not suited for pedestrian trajectory and activity modeling.

Last year, we set off to collect a dataset with different modality in China's road environment, also we intent to tailor our dataset toward pedestrian related tasks, to further facilitate further study.

In section 1.1, multiple sensory inputs parameters are introduced, and data collection protocol are explained. In section 1.2, we will elaborate our data cleaning plan and introduce our data annotation process. In section 1.3, a novel contrastive representation learning method exploiting multi model sensory input in constructed, we were able to reduce manual labeling by leveraging redundant modalities. Finally, in section 1.4 we will show our data collection results.

1.1 Sensory Input

In order to lessen data annotation workload and enable the downstream model to construct a much comprehensive understanding of neighboring environment, we collect multiple sensory modalities including Visual RGB signal, Long-wave infrared thermal signal, LiDAR 3D point cloud depth signal, Microphone array 3D audio signal and telemetry signal including GLONASS and BEIDOU position information and kinetic features collected by IMU(Inertial Measuring Units).



a. RGB

b. LWIR

c. Point Cloud from LiDAR

d. Frame of 5 Channel from Microphone array

Figure 1: Raw sensory input

(1) Visual Signal

In our dataset we collected two set of visual signals with different camera from different location, which enable us to evaluate the impact of different camera has toward downstream tasks, also in the field of single camera depth estimation, the height of camera also has fundamental influence on the terminal performance. First camera is mounted on top of car roof which has a FOV of 60 degree emitting 25fps 1920*1080 resolution footage. Second camera is a GoPro Hero 8 mounted on top of engine hood alongside the LWIR camera for more accurate alignment, the camera is set to 25fps on Narrow which has a FOV of 60 degree. The raw camera footage of example frame is shown in Figure 1-a.

(2) Long-wave Infrared (LWIR) Thermal signal

The most significant signal of pedestrian is their thermal signal**Error! Reference source not found.**, a FLIR VUE PRO 640 camera are set along side the RGB camera on top of the engine hood. The camera are able to detect long-wave(7.5 - 13.5 μm) infrared signal emitted by different heat source with onboard uncooled VOx microbolometer. In our configuration the camera has a FOV of 25° *19° outputting a 9fps 640*512 resolution footage(the framerate is limited to 9fps due to device export control).

(3) LiDAR 3D Point Cloud Signal

In order to obtain a comprehensive depth information, a Velodyne HDL-64E LiDAR is deployed to collect point cloud data. In our experimental setup the lidar takes 1.9 million 3D points per second with a 26.9° vertical FOV. Each point has a azimuth resolution of 0.08° and about 0.4° vertical resolution with maximum distance of around 120m.

(4) Microphone Array 3D Audio Signal

The behavior of pedestrian is heavily affected by the sound of the environment, when electric vehicles approach pedestrian, a mass majority of them report to be frighten. Which recently leads to related regulation about EV emitting sounds. In order to capture sound information of a specific spot, we utilized a open-sourced microphone array. The array has 4 digital microphones aligned in a cross, which each has a omnidirectional sensitivity of -26 dBFS, and overloads at 120dBSPL. The SNR of our mics is 63dB. In our experimental setup all 4 channal of single channel input are collected, and another channel with improved voice quality(far field sound enhancement and de-reverberation) processed by the onboard XVF-3000 chip from XMOS is stored as Ch0**Error! Reference source not found.**.

(5) Positional Signal and Kinetic Signal

The positional signal is collected with GLONASS+BEIDOU RTK system, for better accuracy only best position result during each 100ms. And also the acceleration information are collected with on board IMU.

1.2 Data Processing and Cleaning

i. Data Preprocessing

(1) Sound Source Localization

In our data-preprocessing pipeline, we utilized SRP-PHAT-HSDA method**Error! Reference source not found.** to do sound source localization(SSL) and sound source tracking(SST). The block architecture are shown in Figure 2.

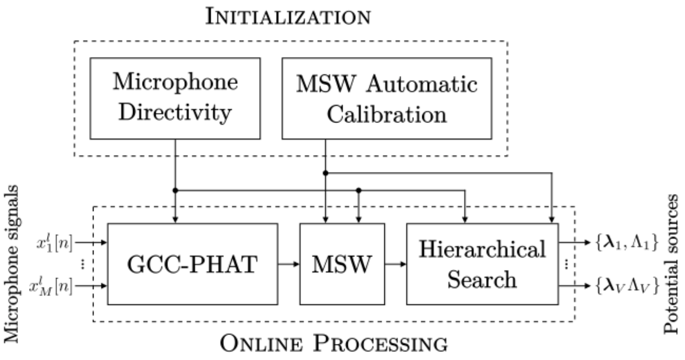


*Figure 2 Framework of sound source localization algorithm**Error! Reference source not found.**.*

The underlying mechanism of SRP-PHAT**Error! Reference source not found.** is to search for V potential sources for each frame over a discrete space. For each potential source, the computed GCC-PHAT frames are filtered using a Maximu Sliding Windows (MSW)**Error! Reference source not found.**. The sum of the filtered GCC-PHAT frames for all pairs of microphones provide the acoustic energy for each direction on the discrete space, and the direction with the maximum energy corresponds to a po- tential source. To further reduce SRP-PHAT computations, and main- tain a high localization accuracy regardless of the micro- phone array shape, SRP-PHAT-HSDA adds Microphone Directivity (MD), Maximum Sliding Window Automatic Calibration (MSWAC) and Hierarchical Search(HS) for better performance.

The localization and tracking result are shown in Figure 3.

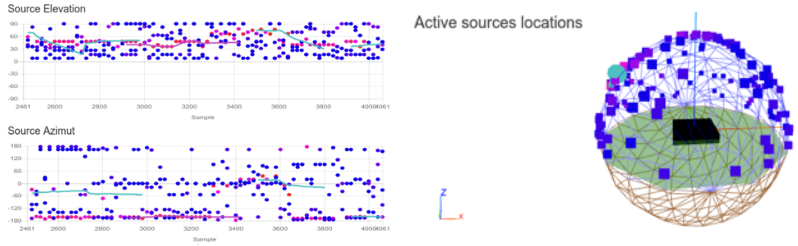
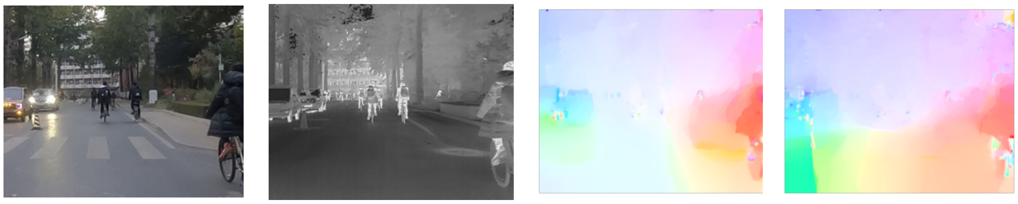


Figure 3 Sound source localization results.

(2) Optical Flow Extraction

In order to obtain motion information between frames and facilitate the representation learning method described in the following sections, we extract TV-L1 optical flow from both RGB image and IR image**Error! Reference source not found.**. In order to ensure consistency and improve estimation speed, we used TV-L1 CUDA GPU implementation provided by OpenCV 3.4.2. The estimation result are shown in Figure 4.

Frame 002936(RGB)



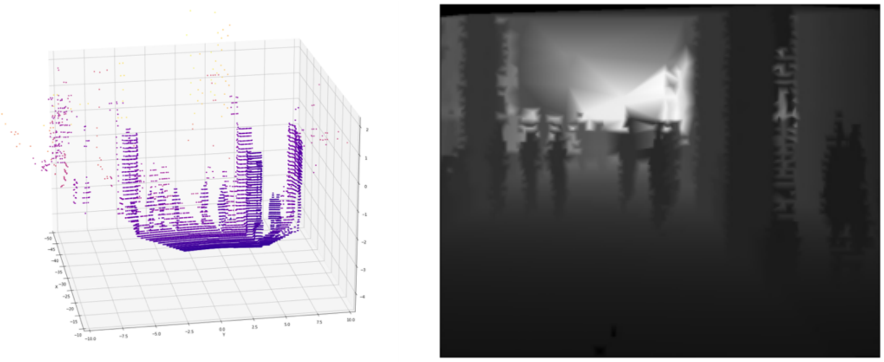
a.RGB b.IR c.Flow-IR d.Flow-RGB

Figure 4 Images extracted optical with TV- *and flow L1*

(3) Depth Map Generation

After aligning point cloud information with our camera, we are able to convert our point cloud information into depth image with gridding and interpolation. Related points are projected into camera space and after linear interpolation, the projected point cloud and depth map are shown in Figure 5.

Frame 008300 (Depth)



a. Point Cloud b. Corresponding Depth map

Figure 5 and *Point cloud* *reconstructed depth map.*

ii. Data Annotation

In our dataset, 2D Tracking Bounding boxes are annotated in the aligned camera space. Currently, there are all together 10625 frames in our dataset, to reduce labeling cost only part of the dataset is labeled by hand, and each target has a unique used for tracking and a class label chosen from 13 road objects. Our interactive labeling environment allows us to dynamically choose key frame for different target and the environment will automatically carry out interpolation through time to fill in other frames. Among which 2432 frames with at least one target are annotated, which contains 16729 hand annotated bounding boxes. Annotation result and 13 class labels are shown in Figure 6.

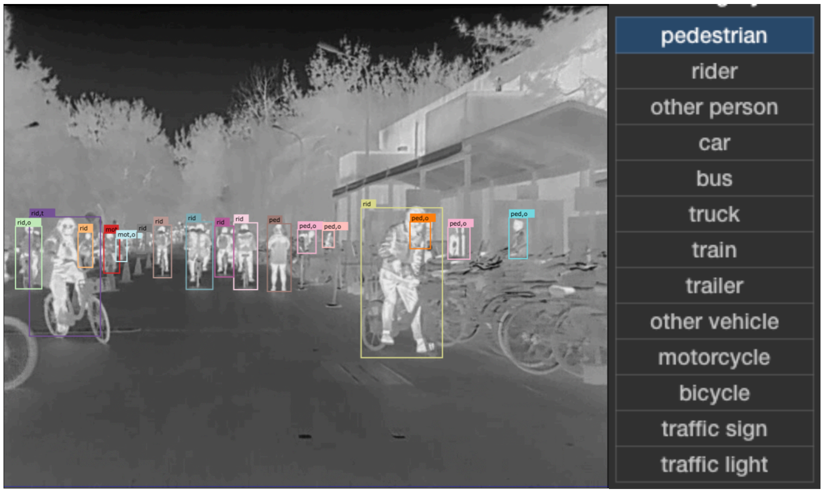


Figure 6 Labeling result and label classes.

1.3 Contrastive Representation Learning

Multi-modal data has a variety of underlying correlation that we can exploit. In this part we intend to use a recent trend of contrastive learning in representation learning field to reduce the human labor needed for constructing a dataset. Traditionally there are three main paradigms in machine learning: supervised, unsupervised and reinforcement learning. For the last decade or so, as a result of fast developing neural network architecture and its strong inference ability, supervised learning is primary choice for training a model for almost every tasks. Yet the supervised paradigm has a inevitable defect, it relies heavily on expensive and hard to get human annotations. Also there are debate about is it possible for it to exceed human performance.

Contrastive learning has recently received interest due to its success in self-supervised representation learning**Error! Reference source not found.**. In this section, we created a novel self-supervised contrastive learning method to train our representation network and only use limited hand-labeled data for downstream tasks such as detection and tracking.

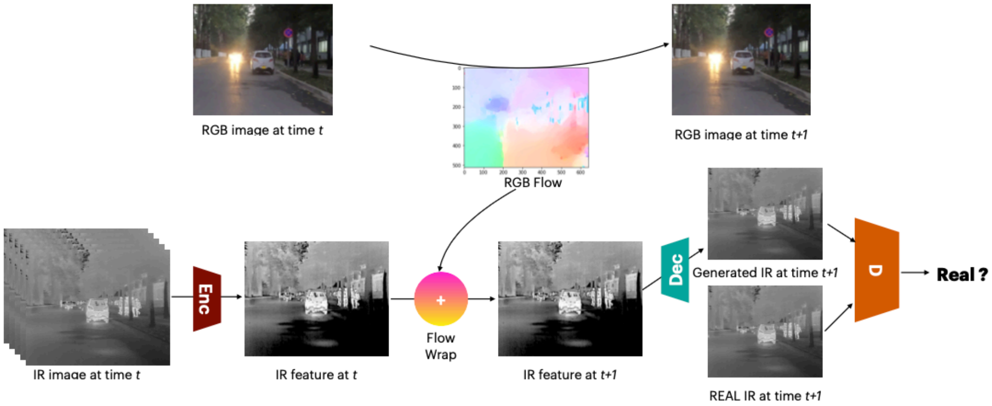


Figure 7 Framework of flow tracked contrastive representation learning

The framework of our proposed contrastive learning method are shown in Figure 7, where we first extract a flow information from a base model, in this case, RGB model. And we construct an Encoder network on other models(in this case, IR model) that has the same output dimension as the target model. The outputted feature from time T is then wrapped with the extracted flow with the intention of getting the feature from time T+1. To validate this, a discriminator is constructed on top of extracted features (One transformed from time T, the other obtained from T+1). By minimizing the distance between two features, we can constrain the information both temporally and spatially.

Traditionally, contrastive learning projects multiple view/modalities into same space, and by minimizing the distance of different view from same scenario, maximizing the distance of view from different scenario. The invariant representation of current scenario is extracted. By doing so, different information of different modality are usually discarded and only identical information are retained.

In our framework, only the spatial and temporal information are constrained with optical-flow, and different information from different modality are retained by reconstruction. As we can see in Figure 8. If the flow is consistent with the feature extracted, the error map will display no difference. If the feature is not consistent with optical flow, the error will display salient differences.

Original Image	Reconstructed Image	Consistant Flow	Error Map

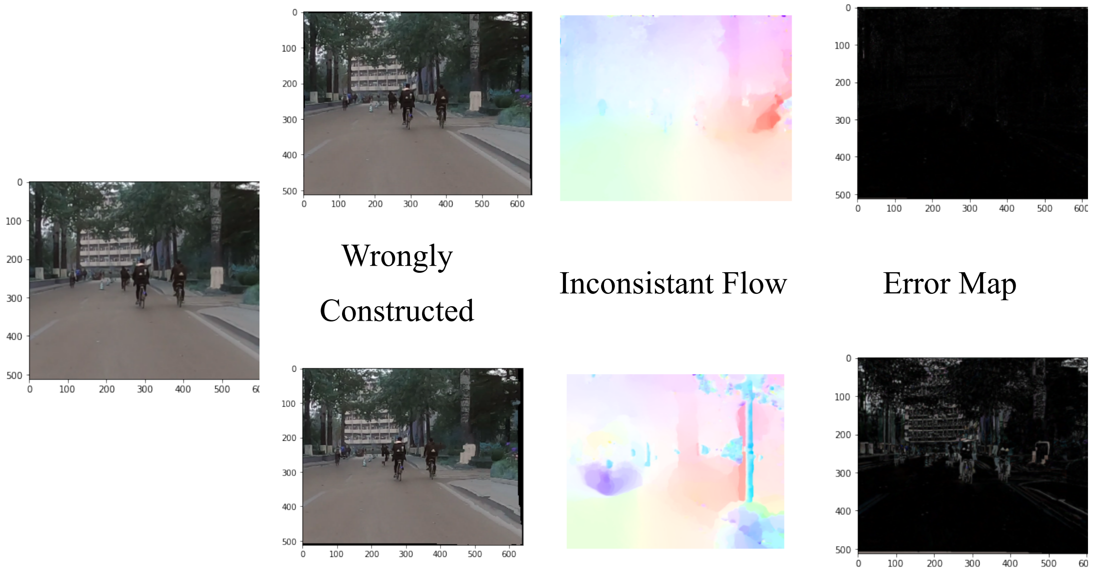


Figure 8 flow wrap result of the same image and their error map.

1.4 Dataset Result

Our final dataset contains 11K frames including 17K hand labeled targets in 3K image frames. Overall size is 55GB, with 49GB of temporal aligned and transposed point cloud data, 1.8GB of wav audio file from all 5 channel, 2.4GB of un-trimmed visual image from RGB camera, 451MB of aligned RGB image, 871MB of extracted TV-L1 image, and 167MB of extracted spectrogram files.

Among 17K labeled targets, there are 5053 labels of pedestrian, 5475 labels of bicycle riders, 2264 labels for cars, 49 labels for bus, 1233 labels for motorcycle, 321 labels for traffic sign, and 477 labels for other vehicles.

2. Dataset Baselines

Extracted feature are tested on detection algorithm to verify the feasibility of our contrasitive learning method. We adopted Faster-RCNN**Error! Reference source not found.** as our decector. The averaged precision of IoU(Intersection over Union) above 50% are shown in Figure 9. Where x-axis are lablels used in training, y-axis are precision. As we can see in both IR modal and RGB modal, the unsupervised pretraining methods significantly improves the detection result over training from scratch.

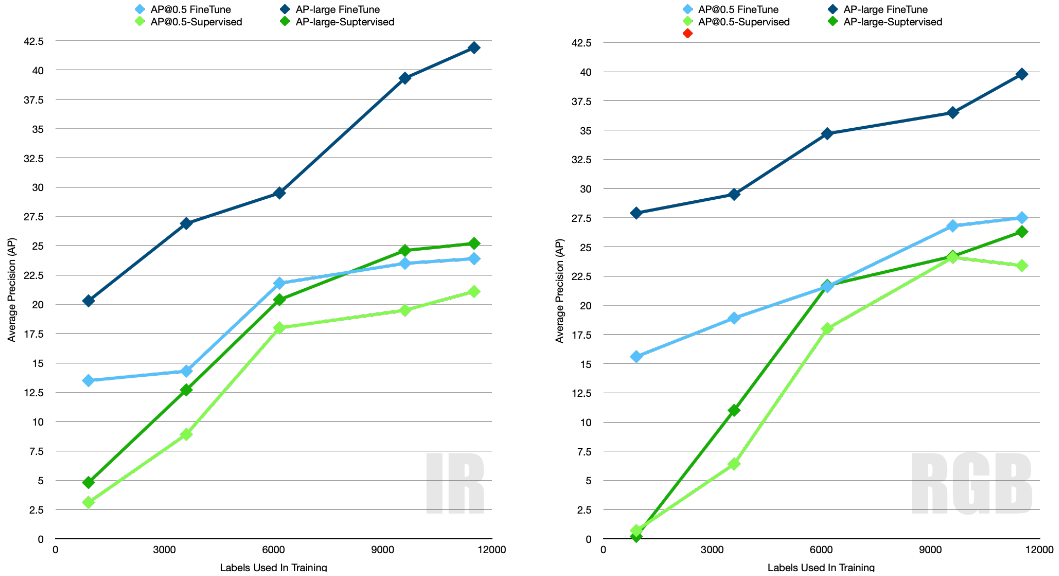


Figure 9 Detection accuracy of two modals over different training set.

Visualization of detection are shown in Figure 10. Where red box represent detected result and green box represent annotated ground truth result.

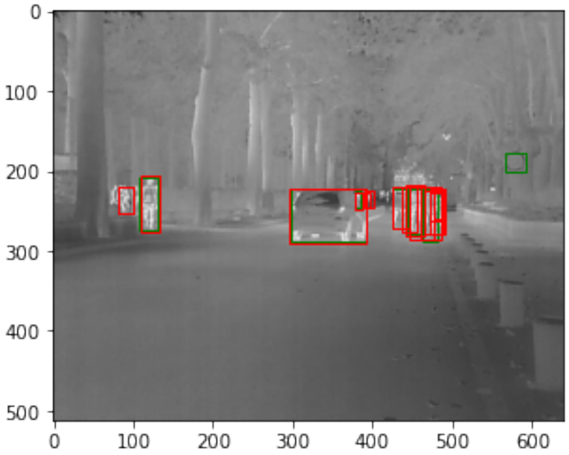


Figure 10 Detection result on our dataset