Cross-Modal Inference of Acceleration Readings from Visual Inputs

Taqiya Ehsan (te137)
Multimodal Learning for Sensing Systems
December 20, 2023

***Abstract*—This research tackles the challenges in person re-identification (ReID) by proposing a cross-modal inference pipeline that integrates visual and inertial measurement unit (IMU) sensor data. Traditional ReID methods relying solely on visual features face limitations in diverse environmental conditions. The introduced approach demonstrates increased resilience to variations in appearance by fusing data from multiple modalities. The study focuses on person-mobile device ReID, mapping individuals in video streams to IMU data from mobile devices. Rigorous testing on different variations of Long-Short Term Memory (LSTM) network achieves up to 100% matching accuracy, emphasizing the method's effectiveness. The proposed pipeline holds promise for real-world applications, particularly in assistive technologies, showcasing the potential of cross-modal inference for enhanced accuracy and efficiency. The poster complementing this paper can be found in this link https://tinyurl.com/mt5ekxsc and a recording of the presentation can be found here https://tinyurl.com/ynuw24ys.**

***Index Terms*—Person re-identification, cross modal inference, multimodal learning, inertial measurement unit (IMU), assistive technologies.**

I. Introduction

Person re-identification (ReID) poses a fundamental challenge within computer vision, striving to match images of the same individual across diverse cameras or temporal instances amidst fluctuating conditions. Traditional ReID techniques, reliant on visual features, prove susceptible to variations in appearance, lighting, pose, or occlusions [1]. A potential remedy involves harnessing cross-modal translation to overcome these constraints by integrating supplementary insights from alternative modalities. Among these, cross-modal inference utilizing inertial measurement unit (IMU) sensors emerges as a solution capable of augmenting the precision and efficiency of ReID systems. IMU sensors excel in capturing distinctive motion features specific to each person, exhibiting greater resilience to changes in appearance and environmental conditions. Moreover, the widespread availability of IMU sensors in mobile devices renders them a pragmatic and promising choice for real-world ReID applications. Notably, cross-modal person ReID holds substantial implications for practical scenarios, such as identity verification in access control systems [2] and providing real-time guidance for the visually impaired in smart streetscapes [3].

In juxtaposition to their susceptibility to challenging environmental factors and sensitivity to variations in appearance, lighting, pose, and occlusions, conventional approaches to ReID frequently revolve around acquiring modality-specific or modality-shareable features. This results in an array of inherent limitations. Firstly, modality-specific features lack generalizability across different modalities. Secondly, the learning of modality-shareable features often entail complex processes such as adversarial training or knowledge distillation, demanding substantial datasets. Compounding these challenges is the predominant reliance of existing ReID methods on image and video data, neglecting the temporal information inherent in data sequences from other modalities. The approach presented in this study aims to overcome the shortcomings of current methodologies by employing a cross-modal inference pipeline to integrate visual and IMU sensor data for person ReID. The proposed pipeline matches people by comparing predicted IMU readings from visual inputs to readily available IMU data. The introduced approach paves the way for novel applications of person ReID in fields like assistive technologies, particularly leveraging the ubiquity of IMU sensor data.

This paper is organized as follows. Section II discusses contemporary studies related to this line of work. Section III sets up the high level objective and approach to person ReID addressed in this work, outlining the overall pipeline. Section III delves deeper into the methods employed for the proposed pipeline. This section elucidates on the dataset, selected features, multimodal data patterns, and overall system description. Section IV highlights the experiments conducted to evaluate the pipeline. Section V goes over the results of the experiments and pipeline performance, while Section VI delineates the limitations and future work of the pipeline before drawing final conclusions in Section VII.

II. Objectives

This paper explores person-mobile device ReID through cross-modal inference, employing both visual and IMU sensor data sourced from mobile devices. The primary objective is to devise an innovative methodology that establishes a connection between IMU data and individuals in videos. The aim is to infer an individual's acceleration from video data and synchronize it with data derived from their mobile phones, thereby tackling the challenges inherent in cross-modal recognition and tracking.

We focus on designing a pipeline to map individuals within a video stream with their IMU data readily available from mobile devices. Through the fusion of data from diverse modalities, this proposed approach demonstrates increased resilience to variations in appearance and other factors that might otherwise impair the effectiveness of ReID methods solely relying on visual information. We rigorously test our pipeline as we add more complexities to it from a naive foundation. Our proposed method achieves up to 100% matching accuracy on a simple Sequence-to-Sequence Long-Short Term Memory network, with and without an attention mechanism, trained by leave-one-out method with a semi-supervised contrastive loss auxiliary function.

III. Related Work

The contemporary object detection task in computer vision is primarily approached through two methods: Convolutional Neural Network (CNN) and Transformers. CNN-based models like YOLO, Faster R-CNN, and RetinaNet are commonly employed in real-world object detection due to their efficiency and accuracy [4]–[6]. Transformer-based detectors such as DETR, ViT, and Swin Transformers have gained popularity in research for their potential to outperform traditional CNN-based models in specific tasks [7]–[9]. While CNN-based models are generally faster and more efficient, making them suitable for real-time applications, transformer-based models are computationally heavier but offer advantages in capturing global context in complex scenes. This paper employs a hybrid approach, combining both a CNN model and the LSTM architecture to enhance understanding of complex spatial and temporal relationships in the given task.

Recent research in the object detection application field has focused on ReID in multi-camera systems to enhance tracking and identification accuracy in urban environments. Studies such as Lu et al. [10], Li et al. [11], and Ye et al. [12] propose strategies for ReID across different imaging modalities, effectively transferring and integrating features from both RGB and infrared modalities. Additionally, works by Jeon et al. [13] and Specker et al. [14] introduce trajectory prediction-based approaches for multi-camera tracking, addressing challenges like object occlusions and appearance variation in different camera views. These advancements significantly improve the performance of ReID techniques, offering practical benefits in urban surveillance and smart city infrastructure, where efficient and accurate tracking and identification are essential [10]–[14].

In the field of ReID, utilizing 3D bounding box information offers more comprehensive details compared to traditional 2D bounding boxes, capturing intricate relationships in data. Recent works by Mahdi et al. [15] and Yan et al. [16] implement robust 3D object recognition systems in complex environments. This approach not only enhances object recognition accuracy in real-world scenarios but also provides detailed spatial data, including positions, velocities, and trajectories. Such spatial information is crucial in ReID applications, facilitating a better understanding of object relationships within a scene and significantly improving tracking and identification accuracy, especially in densely populated or highly dynamic environments [15], [16].

Numerous studies [17]–[19] in human gait research have advanced human identification and activity recognition using sophisticated machine learning techniques. In [18], [19], hybrid neural networks are demonstrated to effectively capture both spatial and temporal aspects of human gaits. These findings underscore the significance of comprehending the gait cycle and its practical applications in real-world scenarios. Our research focuses on collecting IMU data from diverse gait patterns to assist the model in analyzing individual motion patterns. This approach not only improves the model's analytical capabilities but also significantly enhances its robustness [17]–[19].

Cross-modal learning is pivotal in machine learning, integrating information from diverse sensory inputs for improved performance. Research by Zhang et al. [20] highlights the effectiveness of using transformers for semantic segmentation in multi-modal scenarios. In ReID, studies integrate features from modalities like RGB and infrared images, enhancing performance in challenging conditions. Models such as ViLT, ALIGN, and CLIP integrate textual and visual data for tasks like image captioning. In our research, we uniquely apply cross-modality learning by aligning inference data from vision modality with IMU data, crucial for recognition and tracking tasks.

IV. Methods

A. Dataset

A custom dataset has been used for training and evaluating the proposed pipeline. The complete dataset comprises of visual and IMU data for four (4) distinct individuals, collected simultaneously i.e. in the same video stream. The following sub-sections detail the specifics of the data collection procedure.

1) Equipments:

a) Cameras:

- **RTSP Camera**: The camera is positioned at a static height of 100 feet, providing an overview of the parking lot with a field of view spanning 180 degrees. Operating with the Real-Time Streaming Protocol (RTSP), it is set up to continuously stream video in real-time to a designated server, where data is stored and analyzed. The camera boasts a resolution of 1280x720 pixels, capturing content at a rate of 60 frames per second (fps).
- **GoPro Cameras**: Two GoPro cameras are deployed to capture diverse angles and perspectives of subjects within the parking lot. These cameras are configured to record video at a resolution of 1280x720 pixels and a frame rate of 60 frames per second (fps). Each camera has a field of view spanning 180 degrees. They are securely mounted on stationary poles, with one positioned at a height of 100 feet and the other at a height of 80 feet.

b) Mobile Phones: Participants are directed to install a publicly available data collection application, namely Sensor Logger, on their smartphones. This application is designed to capture accelerometer, sampling information at a rate of 100 Hz.

2) Data Collection: The experiment is carried out within a controlled environment, simulating a city streetscape within the parking lot. Data collection is synchronized across the RTSP camera, GoPro cameras, and participants' mobile phones. The cameras record video footage, while the mobile phones gather accelerometer data.

Participants are instructed to follow predefined paths and execute various specific actions and walking patterns while navigating the simulated city streetscape. These actions are crafted to replicate common scenarios and behaviors encountered by pedestrians in urban environments. The following is a list of actions and walking patterns participants were guided to perform:

- **Normal Walking:** Participants were asked to walk at their usual pace and follow a designated path within the streetscape. This action captured baseline walking behavior, including variations in walking speed, direction changes, and interactions with other virtual pedestrians.
- **Crosswalk Interaction:** Participants encountered crosswalks with traffic signals. They were instructed to obey the signals and cross the street when the pedestrian signal was green. This action allowed us to observe how participants responded to traffic conditions.
- **Smartphone Interaction:** At specific points along the path, participants received text messages on their smartphones. They were asked to read and respond to these messages while walking. This action aimed to capture how smartphone distractions affected gait and posture.
- **Obstacle Avoidance:** Participants encountered obstacles, such as pedestrians walking in the opposite direction, parked bicycles, or construction barriers. They were instructed to navigate around these obstacles while maintaining their walking speed. This action assessed participants' ability to adapt to changes in their path.
- **Stop and Wait:** Participants encountered virtual bus stops and benches. They were asked to stop, wait, and pretend to interact with their smartphones as if they were waiting for transportation. This action simulated scenarios where pedestrians pause during their journeys.
- **Street Crossing:** At designated street intersections, participants were required to wait for the traffic signal to change, cross the street when safe, and then continue walking. This action observed how participants interacted with vehicular traffic and other pedestrians at intersections.
- **Sudden Distraction:** Participants experienced unexpected distractions, such as a loud sound or a sudden event (e.g., a vehicle honking). They were instructed to react naturally to these distractions while walking. This action assessed their ability to respond to unexpected events while navigating the urban environment.

3) Data Synchronization: To guarantee data synchronization, a visual cue, specifically a hand movement with the mobile phone in hand, is employed at the commencement of each recording session. This cue serves to align video frames across the three channels and synchronize them with timestamps in the sensor data during the post-processing phase.

4) Experiment Duration: The data collection process lasted for 1 hour, during which participants are recorded performing various actions and walking patterns.

B. Feature Engineering

Inferences from the video data are used as input features of the proposed pipeline. In order to infer features from the visual data, the video stream has been pre-processed frame-by-frame. Moreover, we slice out segments of the video where even one of the individuals is absent (out of camera purview). We ensure the start and end timestamp for data collected for each individual is consistent, and each individual has an equal distribution of feature time steps.

1) Perspective Transformation: The first step we adopted in feature extraction from the video stream is transforming the 2-dimensional camera view into a top-down bird's eye view. To do so we implement the concept of perspective projection using a homography matrix [21]. We use the idea of perspective transformation to project a bird's eye view of the parking lot used for data collection and then cast the trajectories of our individual's movements onto the top-down map of the lot.

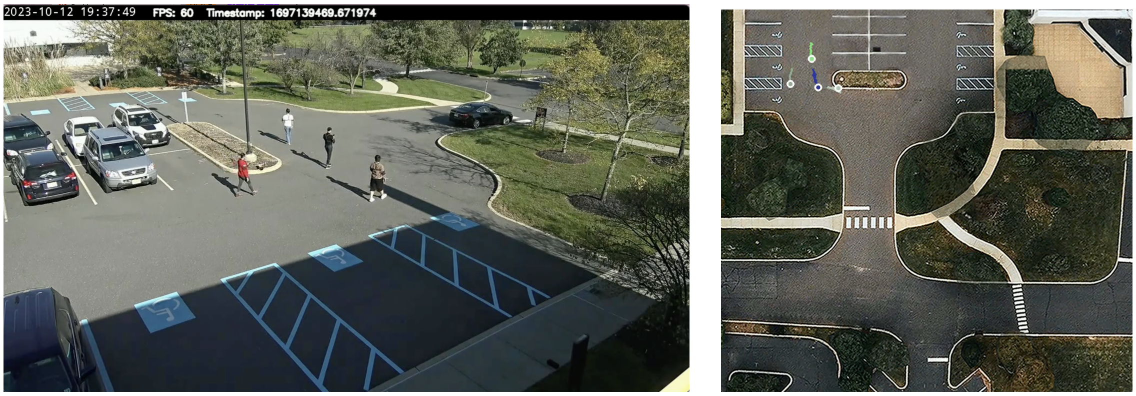
We segment the parking lot in the camera view into nine (9) different regions and map each region to a static top-down image of the same. This enhances the robustness and preciseness of the transformed view. We implement the SIFT algorithm with RANSAC to achieve the matching [22]. From the bird's eye view, each individual is a moving dot on the screen with the color of the dot corresponding to the color of the bounding box around them in the camera view.

This step provides five (5) distinct features: positions (x, y) of the camera, coordinates (x, y) of the individual from bird's eye view, and the trajectory (θ) of their motion.

2) Pedestrian Detection and Tracking: The next step of our feature engineering pipeline is detecting the individuals and tracking their movements through the frames. This is made relatively straightforward by the implementation of YOLOv8 [23] and OC-SORT [24] algorithms. OC-SORT is a tracking algorithm, which if used in conjunction with the latest version (v8) of the state-of-the-art object detection model, YOLO is able to track each individual's movements through the frames as well as assign a unique track ID to each individual that remains consistent throughout the video stream.

This step provides four (4) unique feature inferences from the video stream: coordinates of the 2D bounding boxes (x_1, y_1, x_2, y_2) around the individuals at every time step.

3) 3D Bounding Box Generation: We use the inferences from the previous two steps to generate three dimensional bounding boxes around the individuals. These bounding boxes are based upon the two dimensional bounding boxes that YOLO generates around detected objects. We obtain a total of sixteen (16) features from the eight (8) vertices of the 3D bounding boxes.



Left: Camera view. **Right**: Bird's eye view

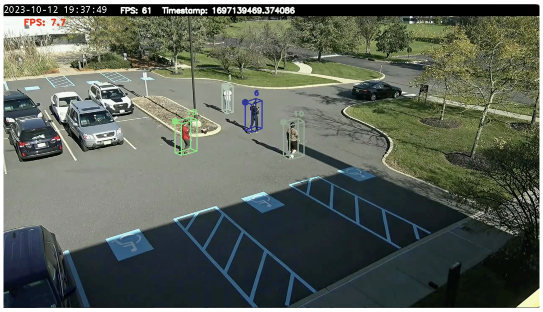


Fig. 1: 3D Bounding Boxes

In order to generate the 3D bounding boxes, we make use of the trajectory (θ) of motion of each individual obtained from the perspective transformation step as well as the 2D bounding box coordinates obtained from YOLO. For each tracked individual, we calculate the bird's eye view bottom coordinates, transform them to the camera view, and construct a 3D bounding box around the individual. The LOWESS (Locally Weighted Least Squares Regression Smoother) [25] algorithm is used to fit a linear regression and obtain orientation of the bottom coordinates in bird's eye.

C. System Description

The objective of this study is framed like a simple multivariate time series forecasting problem. The 25 features inferred from the collected video stream are concatenated into input data. In addition, three (3) acceleration features (x, y, z) from the sensor readings of the app are used as labels of the time series. The goal of the pipeline is to predict the 3 acceleration features given the 25 features inferred from the video stream and map the input features to the correct person (track ID) based on the predictions.

Total duration of video stream used throughout our experiments is approximately 5 minutes (360 seconds) for less computational complexity. For each person (track ID), we segment batches of 6 seconds (180 frames) to train the model. The training data is split into sequences of (92, 180, 25) for the input and (92, 180, 3) for labels to mirror the format (samples, time steps, features). The training is done following the leave-one-out method. All the models are implemented with a Keras backend.

Long-Short Term Memory (LSTM) Network: We implement four different variations of a Long-Short Term Memory (LSTM) network as elaborated on below. We train and test each variation in the same leave-one-out method: training on three (3) track IDs and testing on the remaining (1) track ID. All models are trained for 50 epochs with the efficient Adam version of stochastic gradient descent and a batch size of 1, without resetting the states for every epoch.

We report the train and test loss of the four (4) models for each architecture for better insights.

a) Stacked LSTM: This set of LSTM models have been defined with 2 hidden layers, each with 100 neurons and ReLU activation. The output layer is dense and has 3 neurons for predicting acceleration readings along three axes one time step ahead. Mean squared error is used as the loss function. The sequences are returned at every layer. This model essentially does forecasting on the input time series.

TABLE I: Train and test loss of Stacked LSTM architecture

Model	Test track ID	Train Loss (%)	Test Loss (%)
LSTM A	4	0.104	2.263
LSTM B	3	0.104	1.223
LSTM C	2	0.056	17.588
LSTM D	1	0.057	3.442

b) Sequence-to-Sequence LSTM model: We implement a set of sequence-to-sequence models using the encoder-decoder LSTM architecture. Each of the encoder and decoder has 1 hidden layer with 100 neurons and ReLU activation. There is a repeat vector layer added between the encoder and decoder. A 3-neuron dense layer is added to every temporal slice of the output for predicting acceleration readings along three axes one time step ahead. The sequences are returned at every layer. Mean squared error is used as the loss function.

c) Sequence-to-Sequence LSTM with Attention: We add an attention [26] layer to the previously built sequence-to-sequence models. Both the encoder and decoder has two LSTM layers. The first encoder LSTM layer has 128 units

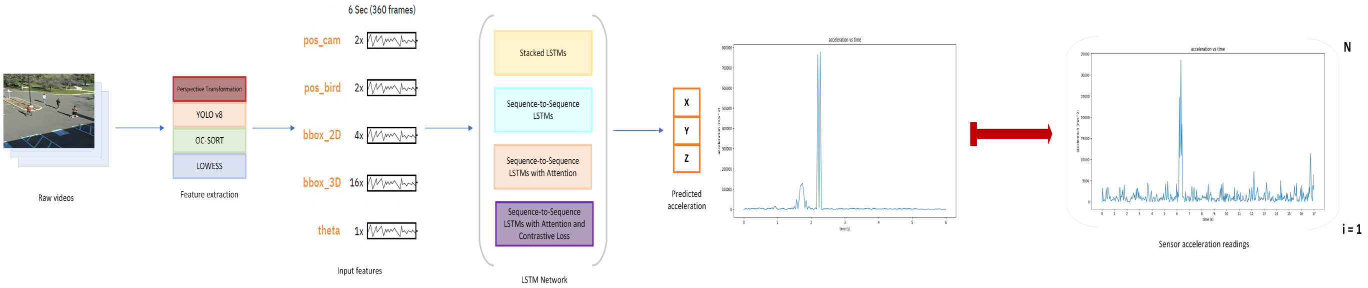


Fig. 2: Basic Pipeline

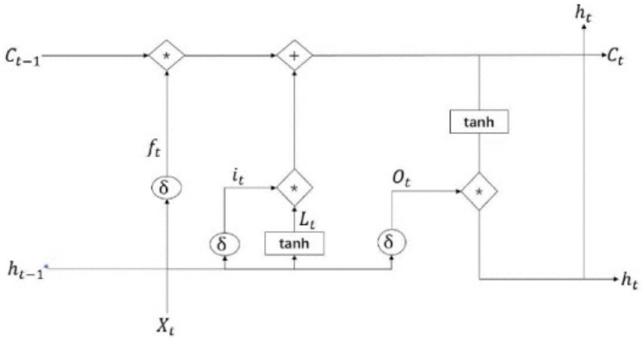


Fig. 3: Long short-term memory network (LSTM) model.

TABLE II: Train and test loss of Sequence-to-Sequence LSTM architecture

Model	Test track ID	Train Loss (%)	Test Loss (%)
Seq2Seq A	4	0.099	2.363
Seq2Seq B	3	0.042	1.182
Seq2Seq C	2	0.059	17.868
Seq2Seq D	1	0.058	3.394

to process the input sequences and return sequences for each time step. The second encoder LSTM layer with 64 units also returns sequences for each time step. The attention mechanism is applied to the encoder's output sequences. We use the Luong-style attention [27] a.k.a dot-product attention available in Keras for this architecture. The attention layer is used to calculate attention weights, and these weights are applied to the encoder's output sequences [28]. The first LSTM layer in the decoder has 64 units and returns sequences for each time step. It takes the attention-weighted encoder output sequences as input. The second LSTM layer in the decoder has 128 units and also returns sequences for each time step. A Time Distributed layer is applied to distribute a Dense layer to each time step in the output sequence. The Dense layer has 3 units, indicating that the model is predicting sequences with a dimensionality of 3 (x, y, z) at each time step. The LSTM layers in both encoder and decoder have ReLU activation. Mean squared error is used as the loss function.

d) Sequence-to-Sequence LSTM architecture with Attention and Contrastive Loss: We modify the loss function and train the same previous set of sequence-to-sequence LSTM models with attention. We use contrastive loss in conjunction with mean squared error as the new function. Contrastive loss was first introduced in [29] as a metric learning loss function. The general formula for Contrastive Loss is:

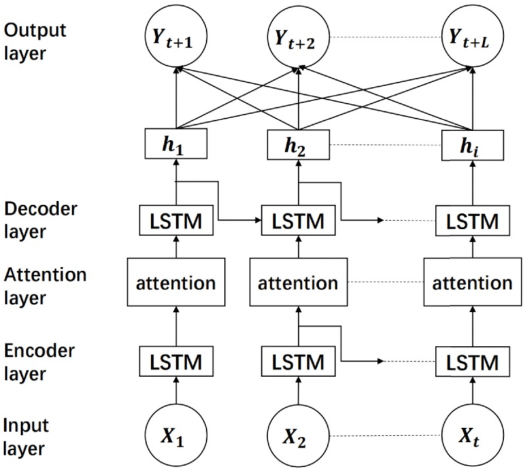


Fig. 4: Structure of LSTM-attention-LSTM model.

TABLE III: Train and test loss of Sequence-to-Sequence LSTM architecture with attention

Model	Test track ID	Train Loss (%)	Test Loss (%)
Seq2Seq_attn A	4	0.098	2.334
Seq2Seq_attn B	3	0.044	1.183
Seq2Seq_attn C	2	0.054	18.145
Seq2Seq_attn D	1	0.057	3.523

$$L(W,(Y,X_1,X_2)^i) = (1-Y)L_S(D_W^i) + YL_DD_W^i \tag{1}$$

The Y term here specifies, whether the two given data points (X_1 and X_2) are similar (Y=0) or dissimilar (Y=1). The L_S term stands for the loss function which should be applied to the output if the given samples are similar, the L_D term is a loss function to apply when the given data points are dissimilar. The D_W term in parenthesis is the similarity (or, rather, dissimilarity) between 2 transformed data points [30].

A version of contrastive learning widely used is dubbed Self-supervised Learning. Self-supervised contrastive learning (SSCL) has demonstrated exceptional efficacy across diverse fields, such as computer vision (CV) [31], [32], [33] and natural language processing (NLP) [34], [35]. This approach leverages unlabeled data to generate both positive (similar) and negative (non-similar) samples, facilitating the learning of representations. There have also been recent studies investigating the effectiveness of SSCL for time-series forecasting. These studies have found that integrating SSCL as an auxiliary objective with Mean Squared Error (MSE) loss yields the best results for time series forecasting tasks in LSTM-based models [36].

Contrastive Loss: We define a custom contrastive loss function to be used with the primary MSE loss pre-defined in Keras. This custom auxiliary loss function computes the pairwise cosine similarities between corresponding elements in the sequences as a measure of how similar the predicted sequence is to the true target sequence for each element. Contrastive loss is measured based on the cosine similarity. The loss is computed as the negative log of the softmax function applied to the scaled similarities. The softmax function is used to obtain a probability-like distribution over the elements in similarities. This contrastive loss encourages the model to increase the similarity for positive pairs and decrease it for negative pairs.

We pass this custom contrastive loss when compiling the model along with MSE. By using this composite loss, the model is simultaneously minimizing the MSE and the contrastive loss during training. The MSE loss is associated with the primary task, while the contrastive loss serves as an auxiliary task, helping the model learn a more meaningful representation.

TABLE IV: Train and test loss of Sequence-to-Sequence LSTM with attention and contrastive loss

Model	Test track ID	Train Loss (%)	Test Loss (%)
Seq2Seq_attn A	4	0.099	2.295
Seq2Seq_attn B	3	0.043	1.161
Seq2Seq_attn C	2	0.057	17.891
Seq2Seq_attn D	1	0.057	3.478

V. Experiments

To investigate the robustness and accuracy of the models, we evaluate all the trained models for every LSTM architecture with video inferences for all individuals. That is, we take the 25 features inferred from the video stream for every individual and predict their accelerations for evaluating each model's performance. For evaluation of the models' performance, we compare the predictions with the ground truth IMU data using Dynamic Time Warping (DTW) distance. The lower the DTW cost between the predicted and ground truth time-series, the more similar the two sequences are i.e. more like they are to be for the same individual (track ID).

VI. Results & Pipeline Performance

Our approach is to match a predicted time series to the track ID whose ground truth time series it (prediction) has lowest DTW distance with. To enhance the robustness of the matching process, we expand our threshold/buffer to k lowest DTW distances.

- The best performing **Stacked LSTM** model is *LSTM D* with test loss of 3.44 and 75% correct matching for a threshold of 3 lowest DTW distance values.

TABLE V: Average accuracy for user mapping to k lowest DTW distances with leave-one-out training of Stacked LSTM models

Model	k = 1	k = 2	k = 3
LSTM A	0%	75%	75%
LSTM B	0%	50%	75%
LSTM C	0%	50%	75%
LSTM D	50%	75%	75%

- The best performing **Sequence-to-Sequence** (Seq2Seq) LSTM model is *Seq2Seq B* with test loss of 1.18 and 100% correct matching for a threshold of 3 lowest DTW distance values.

TABLE VI: Average accuracy for user mapping to k lowest DTW distances with leave-one-out training of Seq2Seq LSTM models

Model	k = 1	k = 2	k = 3
Seq2Seq A	50%	50%	75%
Seq2Seq B	0%	25%	100%
Seq2Seq C	25%	50%	75%
Seq2Seq D	25%	50%	75%

- The best performing **Sequence-to-sequence LSTM** model with **attention** is *Seq2Seq_attn A* with test loss of 2.33 and 100% correct matching for a threshold of 3 lowest DTW distance values.

TABLE VII: Average accuracy for user mapping to k lowest DTW distances with leave-one-out training of Seq2Seq LSTM models with Attention

Model	k = 1	k = 2	k = 3
Seq2Seq_attn A	50%	75%	100%
Seq2Seq_attn B	25%	50%	75%
Seq2Seq_attn C	25%	50%	75%
Seq2Seq_attn D	25%	75%	100%

- The best performing **Sequence-to-Sequence LSTM** model with **attention** and **contrastive loss** is *Seq2Seq_attn_CL D* with test loss of 3.478 and 100% correct matching for a threshold of 3 lowest DTW distance values.

TABLE VIII: Average accuracy for user mapping to k lowest DTW distances with leave-one-out training of Seq2Seq LSTM with attention and contrastive loss

Model	k = 1	k = 2	k = 3
Seq2Seq_attn_CL A	0%	25%	25%
Seq2Seq_attn_CL B	25%	50%	75%
Seq2Seq_attn_CL C	25%	50%	75%
Seq2Seq_attn_CL D	0%	75%	100%

VII. Limitations & Future Work

The input and output data for our collected dataset is out-of-distribution. That is to say, for every similar pair of data points there are more dissimilar data points. This is what causes the contrastive loss function to perform poorly. To combat this drawback, we would need to augment our input dataset with the same inferences but from different transformations of the video stream. Moreover, we only use 5 minutes worth of data in the training and evaluation of these models. Using the entire 1 hour of video stream would result in a more robust and accurate pipeline.

Future work would also include continued introduction of complexities to make the pipeline more robust, reliable, and adaptable. We plan on streamlining the methodology such that a single model can perform video inferences and person-device ReID in one shot in real time.

VIII. Conclusion

This paper proposes a robust methodology connecting IMU data with individuals in videos, with a focus on overcoming challenges inherent in cross-modal recognition and tracking. We investigate several different variations of an LSTM-based architecture with progressively introduced complexities. The overall best performing variation is a Sequence-to-Sequence LSTM with an Attention mechanism. The designed pipeline successfully maps individuals within a video stream to their corresponding IMU data, given some threshold for error, showcasing increased resilience to variations in appearance and other factors. In summary, this paper not only contributes to advancing person-mobile device ReID but also provides a solid foundation for future research in the dynamic field of cross-modal inference, emphasizing the importance of integrating multiple data modalities for more resilient and accurate recognition and tracking of individuals in diverse settings.

References

[1] M. Ye, J. Shen, G. Lin, T. Xiang, L. Shao, and S. C. Hoi, "Deep learning for person re-identification: A survey and outlook," *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 6, pp. 2872–2893, 2021.

[2] G. Amato, P. Barsocchi, F. Falchi, *et al.*, "Towards multimodal surveillance for smart building security," in *Proceedings*, MDPI, vol. 2, 2018, p. 95.

[3] G. Jain, Y. Teng, D. H. Cho, Y. Xing, M. Aziz, and B. A. Smith, "" i want to figure things out": Supporting exploration in navigation for people with visual impairments," *Proceedings of the ACM on Human-Computer Interaction*, vol. 7, no. CSCW1, pp. 1–28, 2023.

[4] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, *You only look once: Unified, real-time object detection*, 2016. arXiv: 1506.02640 [cs.CV].

[5] S. Ren, K. He, R. Girshick, and J. Sun, *Faster r-cnn: Towards real-time object detection with region proposal networks*, 2016. arXiv: 1506.01497 [cs.CV].

[6] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, *Focal loss for dense object detection*, 2018. arXiv: 1708.02002 [cs.CV].

[7] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, *End-to-end object detection with transformers*, 2020. arXiv: 2005.12872 [cs.CV].

[8] A. Dosovitskiy, L. Beyer, A. Kolesnikov, *et al.*, *An image is worth 16x16 words: Transformers for image recognition at scale*, 2021. arXiv: 2010.11929 [cs.CV].

[9] Z. Liu, Y. Lin, Y. Cao, *et al.*, *Swin transformer: Hierarchical vision transformer using shifted windows*, 2021. arXiv: 2103.14030 [cs.CV].

[10] Y. Lu, Y. Wu, B. Liu, *et al.*, *Cross-modality person re-identification with shared-specific feature transfer*, 2020. arXiv: 2002.12489 [cs.CV].

[11] D. Li, X. Wei, X. Hong, and Y. Gong, "Infrared-visible cross-modal person re-identification with an x modality," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 04, pp. 4610–4617, Apr. 2020. DOI: 10.1609/aaai.v34i04.5891.

[12] M. Ye, J. Shen, and L. Shao, "Visible-infrared person re-identification via homogeneous augmented tri-modal learning," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 728–739, 2021. DOI: 10.1109/TIFS.2020.3001665.

[13] Y. Jeon, D. Q. Tran, M. Park, and S. Park, "Leveraging future trajectory prediction for multi-camera people tracking," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2023, pp. 5399–5408. DOI: 10.1109/CVPRW59228.2023.00570.

[14] A. Specker and J. Beyerer, "Reidtrack: Reid-only multi-target multi-camera tracking," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2023, pp. 5442–5452. DOI: 10.1109/CVPRW59228.2023.00575.

[15] M. Rezaei, M. Azarmi, and F. M. P. Mir, "3d-net: Monocular 3d object recognition for traffic monitoring," *Expert Systems with Applications*, vol. 227, p. 120 253, 2023, ISSN: 0957-4174.

[16] J. Yan, Y. Liu, J. Sun, *et al.*, *Cross modal transformer: Towards fast and robust 3d object detection*, 2023. arXiv: 2301.01283 [cs.CV].

[17] W. Hu, X. Xiao, Z. Fu, D. Xie, T. Tan, and S. Maybank, "A system for learning statistical motion patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 9, pp. 1450–1464, 2006. DOI: 10.1109/TPAMI.2006.176.

[18] R. Asokan, V. Pathmanaban, V. Shenbaga Shudhan, and S. Abirami, "Gait based human activity recognition using hybrid neural networks," in *2023 12th International Conference on Advanced Computing (ICoAC)*, 2023, pp. 1–6. DOI: 10.1109/ICoAC59537.2023.10249714.

[19] C. Prakash, R. Kumar, and N. Mittal, "Recent developments in human gait research: Parameters, approaches,

applications, machine learning techniques, datasets and challenges," *Artif. Intell. Rev.*, vol. 49, no. 1, pp. 1–40, Jan. 2018, ISSN: 0269-2821. DOI: 10.1007/s10462-016-9514-6.

[20] J. Zhang, H. Liu, K. Yang, X. Hu, R. Liu, and R. Stiefelhagen, *Cmx: Cross-modal fusion for rgb-x semantic segmentation with transformers*, 2023. arXiv: 2203.04838 [cs.CV].

[21] S. Lee, "Understanding homography (aka perspective transformation)," *Towards Data Science: https://towardsdatascience. com/understanding-homography-aka-perspectivetransformation-cacaed5ca17 (Erişim tarihi: 1 Eylül 2022)*, 2022.

[22] V. Viana, *Sift and ransac algorithms*, Jun. 2020. [Online]. Available: https://www.kaggle.com/code/victorvianaom/sift-and-ransac-algorithms.

[23] Ultralytics, *Ultralytics yolov8 docs*. [Online]. Available: https://docs.ultralytics.com/.

[24] J. Cao, J. Pang, X. Weng, R. Khirodkar, and K. Kitani, "Observation-centric sort: Rethinking sort for robust multi-object tracking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 9686–9696.

[25] J. P. Figueira, *Loess: Smoothing data using local regression*, Jun. 2021. [Online]. Available: https://towardsdatascience.com/loess-373d43b03564.

[26] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, "Attention is all you need," *CoRR*, vol. abs/1706.03762, 2017. arXiv: 1706.03762. [Online]. Available: http://arxiv.org/abs/1706.03762.

[27] M.-T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," 2015. arXiv: 1508.04025 [cs.CL].

[28] X. Wen and W. Li, "Time series prediction based on lstm-attention-lstm model," *IEEE Access*, vol. 11, pp. 48 322–48 331, 2023. DOI: 10.1109/ACCESS.2023.3276628.

[29] R. Hadsell, S. Chopra, and Y. Lecun, "Dimensionality reduction by learning an invariant mapping," Feb. 2006, pp. 1735–1742, ISBN: 0-7695-2597-0. DOI: 10.1109/CVPR.2006.100.

[30] M. Bekuzarov, *Losses explained: Contrastive loss*, Apr. 2020. [Online]. Available: https://medium.com/@maksym.bekuzarov/losses-explained-contrastive-loss-f8f57fe32246.

[31] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning. arxiv e-prints, art," *arXiv preprint arXiv:1911.05722*, 2019.

[32] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*, PMLR, 2020, pp. 1597–1607.

[33] K. Kotar, G. Ilharco, L. Schmidt, K. Ehsani, and R. Mottaghi, "Contrasting contrastive self-supervised representation learning pipelines," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 9949–9959.

[34] T. Gao, X. Yao, and D. Chen, "Simcse: Simple contrastive learning of sentence embeddings," *arXiv preprint arXiv:2104.08821*, 2021.

[35] F. Liu, I. Vulić, A. Korhonen, and N. Collier, "Fast, effective, and self-supervised: Transforming masked language models into universal lexical and sentence encoders," *arXiv preprint arXiv:2104.08027*, 2021.

[36] C. Zhang, Q. Yan, L. Meng, and T. Sylvain, "What constitutes good contrastive learning in time-series forecasting?" *arXiv preprint arXiv:2306.12086*, 2023.