

The logo for the 2024 embedded VISION SUMMIT is centered in a white octagonal shape. The text "2024" is at the top, "embedded" is below it, "VISION" is in large, bold, blue letters with a yellow-to-orange gradient, and "SUMMIT" is at the bottom. The octagon is surrounded by a colorful geometric pattern of overlapping triangles in shades of purple, blue, green, yellow, and orange.

2024
embedded
VISION
SUMMIT®

Improved Navigation Assistance for the Blind via Real-time Edge AI

Aishwarya Jadhav

- Software Engineer, Tesla Autopilot
- Project Sponsor/Mentor,
Carnegie Mellon University



Our Vision

- Given visual sensory inputs, can an AI app predict in real time how a sighted human would react in future?
- And then convey this prediction audibly to assist a blind individual



AI Guide Dog

Motivation

- Car GPS provides coarse directions
- Human brain, sight and motor control provide **fine-grained** actions based on immediate surroundings

- Blind users already use GPS apps for high level directions and their cane + mobility skills for navigating surroundings
- Could we give a real-time AI system for fine-grained directions?





Accurate: Few false positives



Explainable: The high-risk and sensitive setting cannot tolerate black-box decisions



Realtime: Low latencies for continually processing the stream of input video

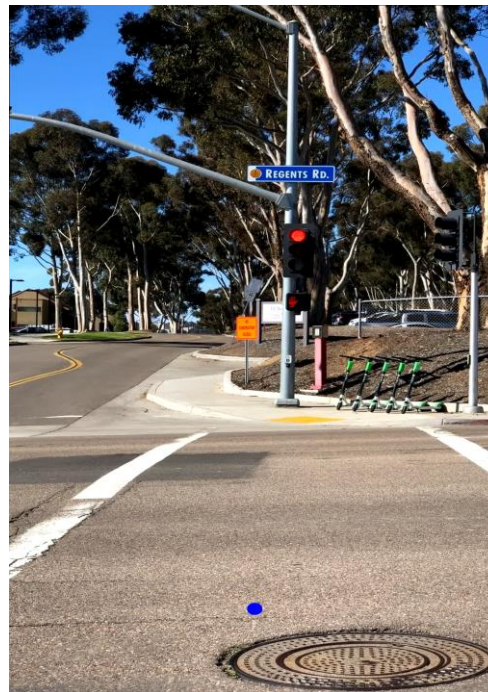


Lightweight: For deployment and use on a mobile app

- **Classification Problem**



- **Regression Problem**



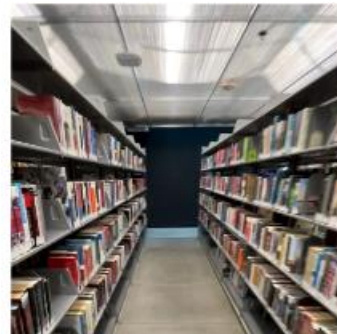
Modeling Strategy 1: Classification

- **Time series prediction problem:** For every frame, predict whether to turn LEFT, RIGHT or keep STRAIGHT in the next 1 second.
- Models: CNNs, ConvLSTMs, PredRNN, multimodal transformers (sensors)

Input	Indoor Availability	Outdoor Availability	Models
Image (Video Frames)	✓	✓	CNN, ConvLSTM, PredRNN, Multimodal
GPS Directions (Intent)	✗	✓	CNN, ConvLSTM
Sensors Signals	✓	✓	Multimodal

Classification Approach — Dataset

- Labels: Left/Right/Front
- Combination of three datasets totaling 7 hours
 - Seattle library dataset
 - Pittsburgh library dataset
 - Grocery store dataset
 - Outdoor dataset
- Dataset configuration:
 - 64x64, 2 fps
 - Black and white
 - 10 frames per sequence
- 5 second clips for training and eval
 - 4 seconds context + 1 sec future prediction ground truth



Seattle Library



Outdoor



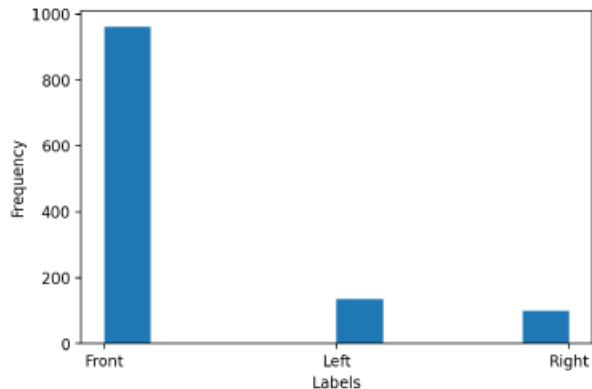
CMU Library



Grocery Store

Loss Functions

- High imbalance in data



- **Weighted loss:**

$$L_{\text{Weighted Loss}}(x, y) = - \sum_{i=1}^{\text{All Class}} w_{y_i} * \log \left(\frac{\exp x_i}{\sum_{i=1}^{\text{All Class}} \exp x_i} \right)$$

w_{y_i} : class weight (inverse of class sample counts)

- **Focal loss:**

$$L_{\text{Focal Loss}}(x, y) = - \sum_{i=1}^{\text{All Class}} \alpha_i * (1 - p_i)^y * \log(p_i)$$

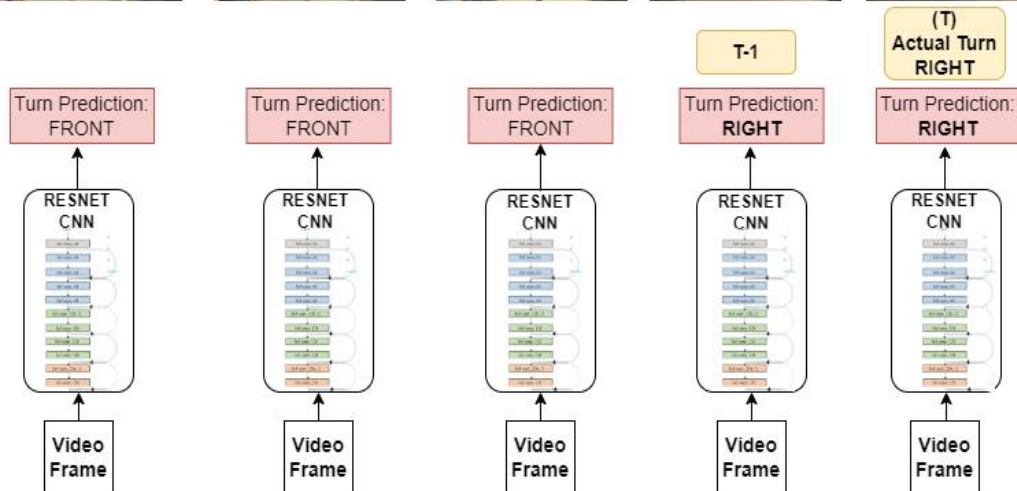
p_i : output from classifier

α_i : class weight (inverse of class sample counts)

- Based on ablations, weighted focal loss gave the best results
- Its focused attention to hard turn examples is quite effective

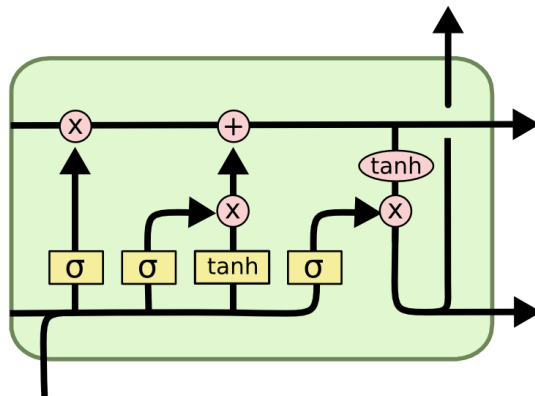
CNN-based Model

- ResNet34 to extract video frame features to detect turn in the near future.
 - Fine-tuning for our datasets
- Pros: State of the art architecture pre-trained on a very a large image corpus.
 - Useful for image feature extractions
- Cons: Considers only point-in-time information without any regard for the temporal aspects of the videos.
 - Additional non-trivial modifications needed to incorporate more signals

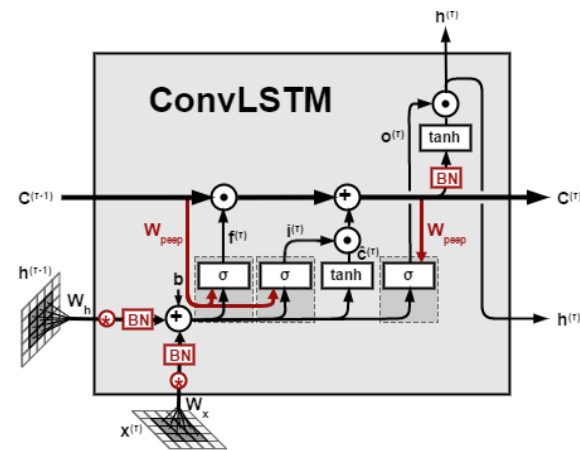


ConvLSTM-based Models

- Type of recurrent neural network for spatio-temporal prediction
- Pros: Predictions are informed by the time-series history of the frames preceding the current frame
 - Easy incorporation of additional signals
- Cons:
 - Need to train from scratch
 - Computationally expensive

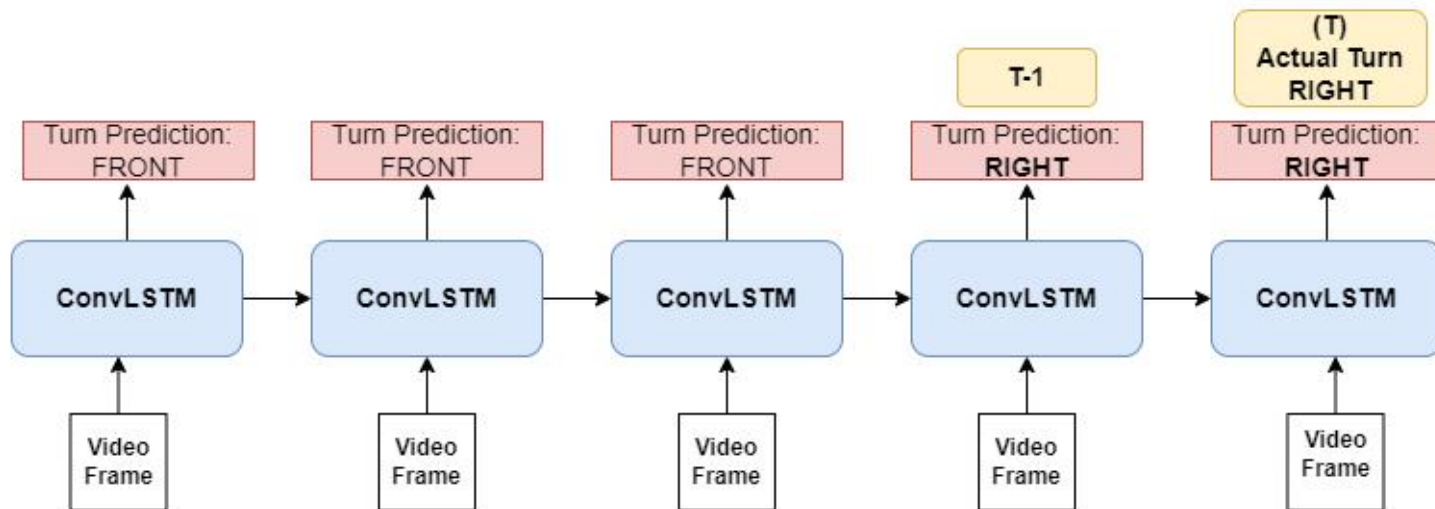


LSTM Block



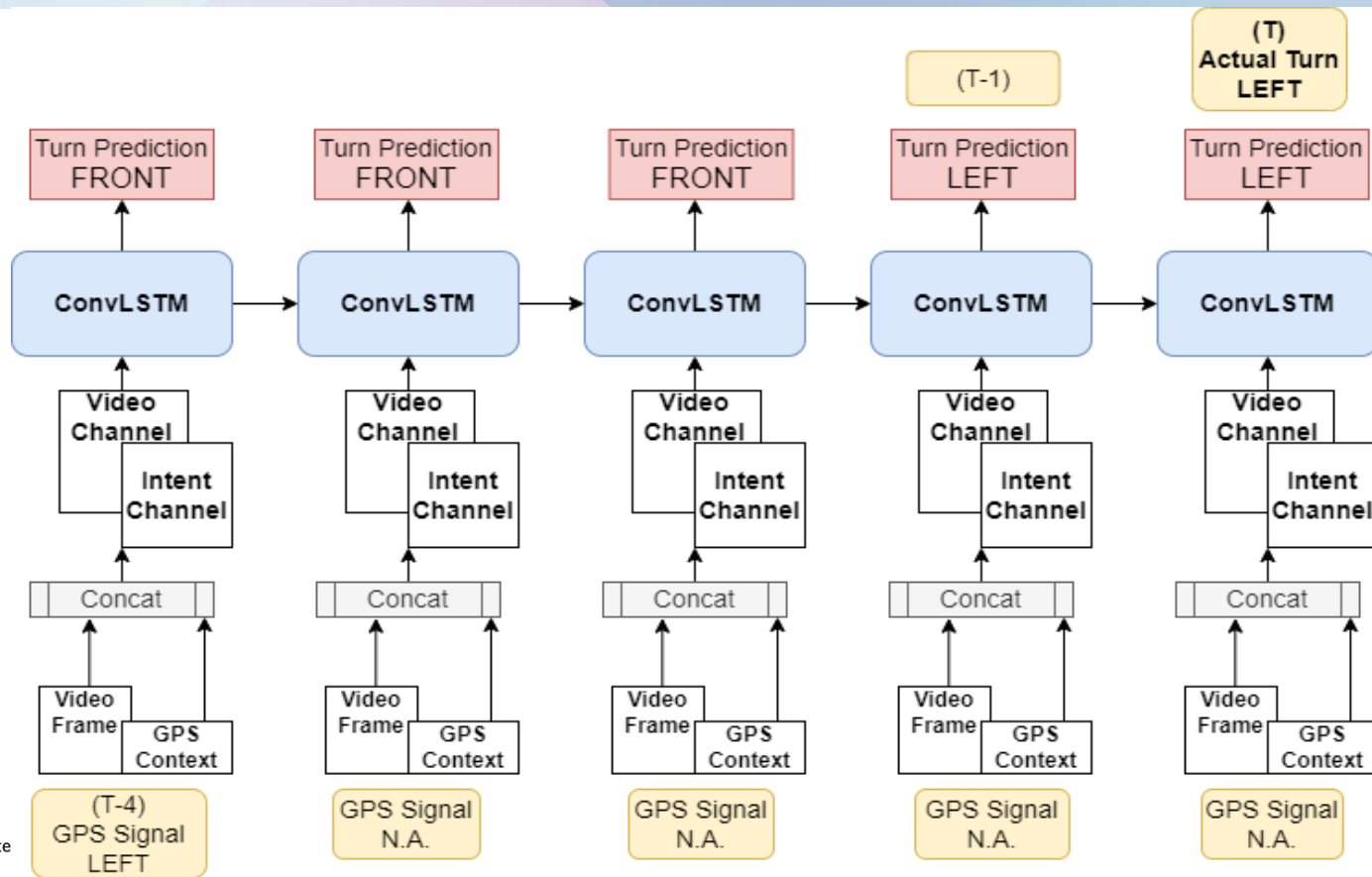
ConvLSTM Block

ConvLSTM-based Models



- **GPS signals** used to determine the path to destination in an outdoor environment.
 - Usually received 2 to 5 seconds before the turn.
 - Serves as **intent** to determine the direction at an intersection.
- Modified problem description: Predict the turn depending on Intent (GPS based directions) and walking speed
 - Walking speed: Determined using egocentric video captured using users' mobile app.
- Indoor setting: GPS signal not available
 - Simulate intent signals to train and compare with other indoor models.
 - Randomly introduce intent specifying direction 2 to 5 seconds before the turn while training.

Modified ConvLSTM with Intent



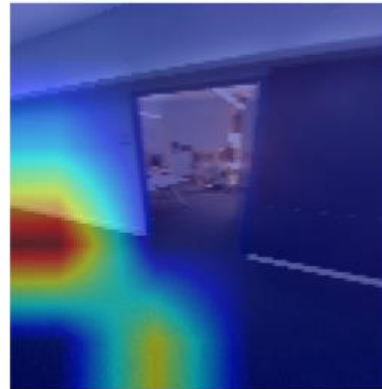
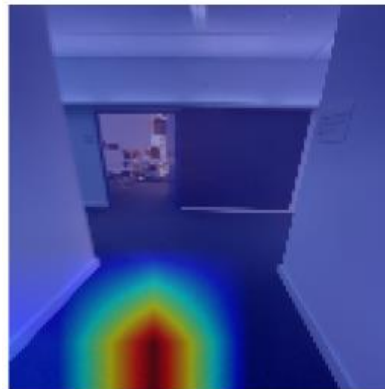
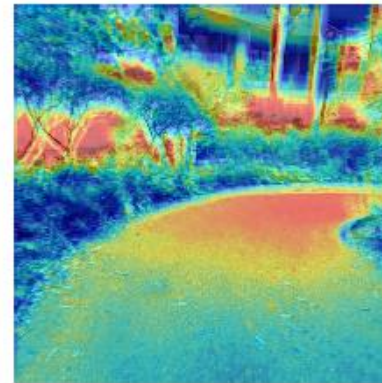
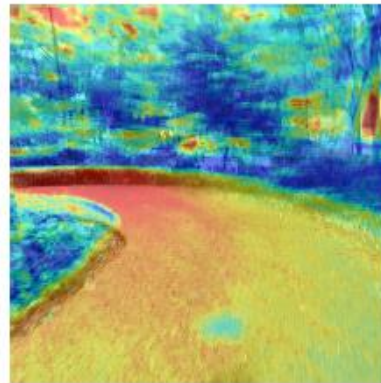
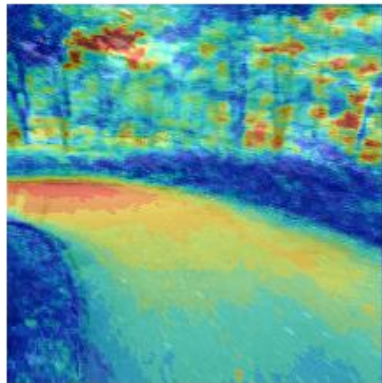
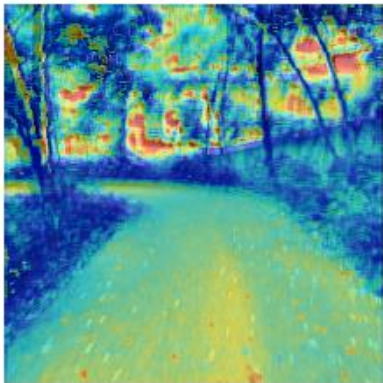
Model	Accuracy	Precision			Recall			F1		
		Left	Front	Right	Left	Front	Right	Left	Front	Right
CNN No Intent	0.608	0.359	0.693	0.356	0.2	0.804	0.331	0.259	0.744	0.343
ConvLSTM Base	0.587	0.386	0.73	0.4	0.355	0.723	0.414	0.357	0.727	0.407
ConvLSTM Video + Gyrometer	0.588	0.354	0.756	0.400	0.331	0.686	0.529	0.342	0.720	0.456
ViT + BERT Video + Gyrometer	0.609	0.348	0.878	0.417	0.475	0.66	0.579	0.402	0.754	0.485
PredRNN with focal loss	0.667	0.549	0.788	0.481	0.460	0.747	0.619	0.501	0.767	0.541
CNN with LSTM + Intent	0.70	0.559	0.767	0.54	0.441	0.804	0.563	0.493	0.785	0.551

Comparison of model performances

- Pre-training on large corpuses greatly benefits the learning process by presenting powerful representations (e.g., CNN and ViT).
- Intent signals help with disambiguation of path.

Results: Explainable Predictions

- GradCAM visualizations from ConvLSTM model



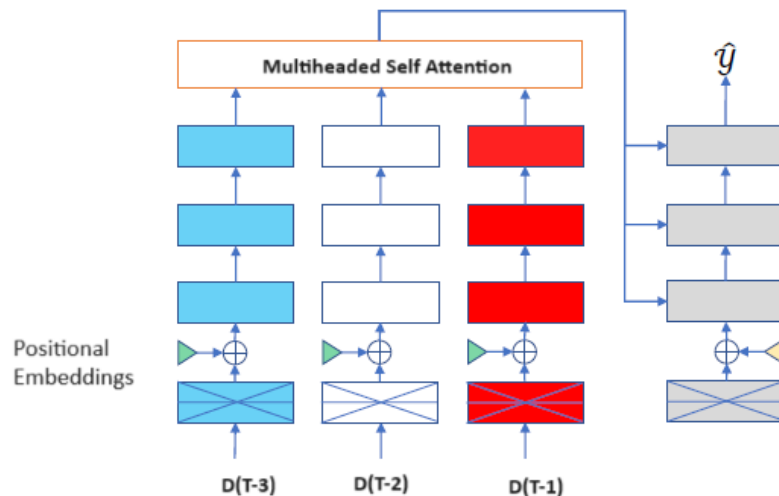
Modeling Strategy 2: Regression (Current Research)

- **Problem statement:** Learn a function $f(Y, Z) \rightarrow Y'$ which takes the past trajectory Y of the camera wearer and contextual cues Z as inputs, and outputs the camera wearer's future trajectory Y' .
 - Contextual information: Scene semantics, depth information, other pedestrian trajectories
 - More fine-grained predictions wrt. position and orientation of camera person
- **Transformers!**
 - Fast and lightweight
 - Easily incorporate multimodal + intent information
- Loss: MSE + L2
- Eval Metrics: MSE

Autoregressive Transformer

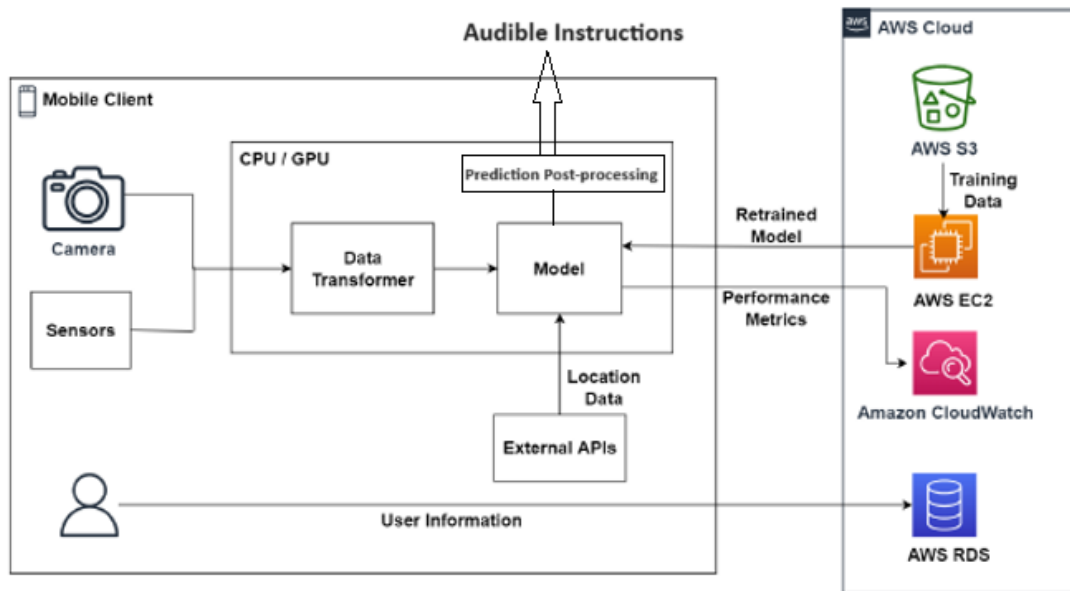
Each timestep encoding captures the following information:

- Camera wearer's trajectory:
 - Formed using the camera position and orientation at each timestep
 - ORB-SLAM3 to generate the ground truth trajectories of the camera wearer
- Semantic scene segmentation masks:
 - PSPNet to generate frame segmentation masks
 - Trained an auto-encoder to convert mask to k-dim vector
- Other pedestrian trajectories: Future work



Deployment

- Deployment considerations:
 - Model quantization
 - Data privacy
- On-device performance metrics:
 - Inference time
 - Memory usage
 - Energy consumption



Mobile App Trade-Offs

Quantization

- 16-bit model offers good trade-off between performance and app size.
- GPU utilization increases as frames rate increases and decreases with quantization.

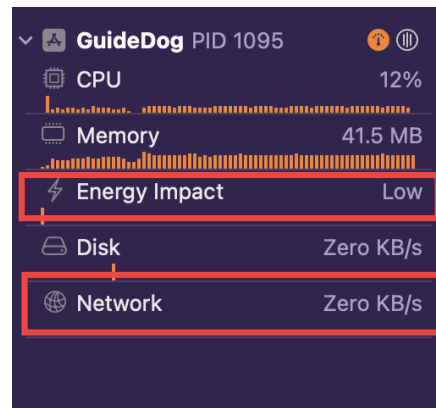
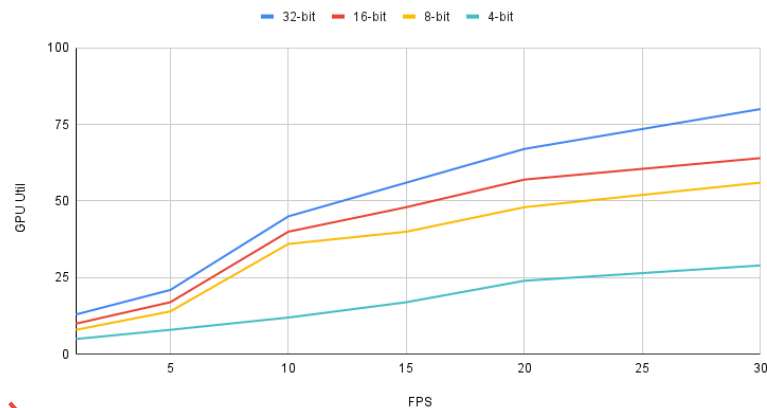
Battery utilization

- Even with 30 FPS (maximum), effect on battery life is not significant.
- In practice, 1 FPS is sufficient given human walking speeds, for best battery usage and to avoid device heating

Data privacy

- All processing and inference is on-device and users' data does not leave the phone

GPU utility with quantisation of model



- Egocentric human trajectory prediction is a fairly unexplored research domain with extremely important applications
 - Most research address autonomous vehicles
 - Or third-person trajectory prediction from egocentric view
- Evolving research in computer vision with better architectures can be applied to this problem to create direct real-world impact
- Major challenges remain to efficiently deploy on edge devices and build trust among users
- Detailed technical presentation by the team: [slides](#)

Thank You!