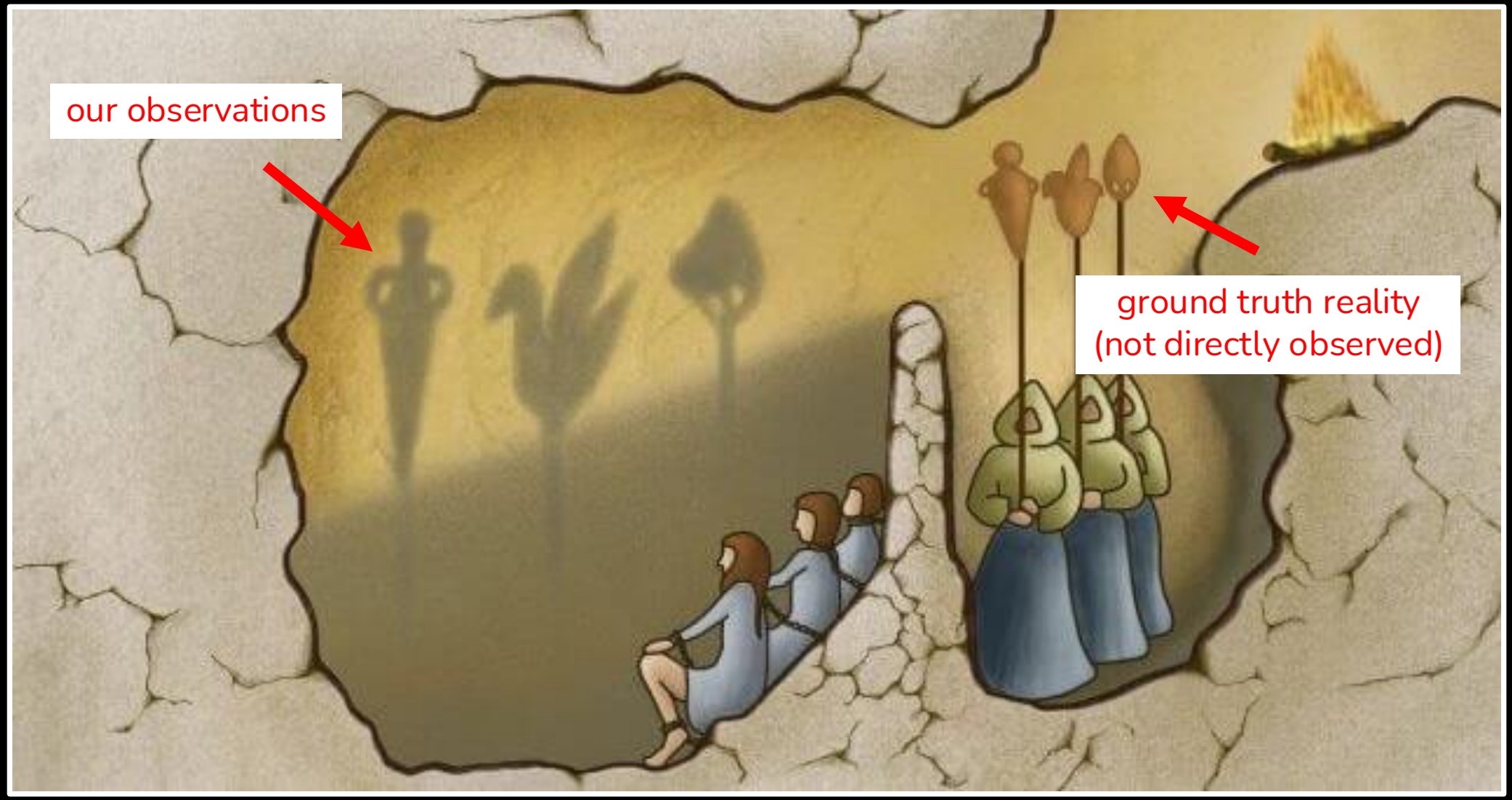
Multi-Modal Foundation Models

Siddharth Somasundaram

Paper Presentation

Babel: A Scalable Pre-trained Model for Multi-Modal Sensing via Expandable Modality Alignment

Allegory of Plato's Cave

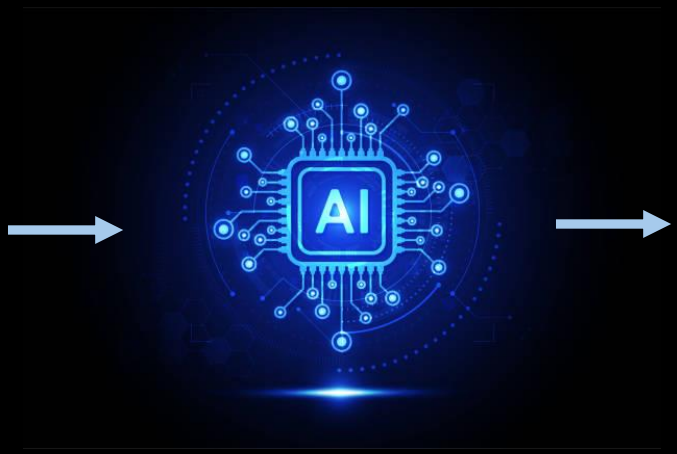


How We Train AI Models

goal is to accurately describe this

color image

sensor



Latent Unobservable World

Observable Projection of the World

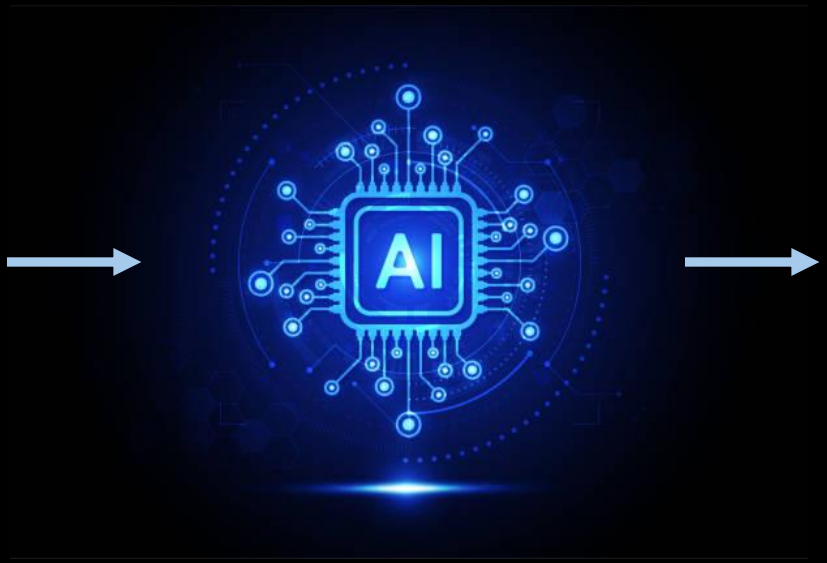
AI Model

Feature State of the World

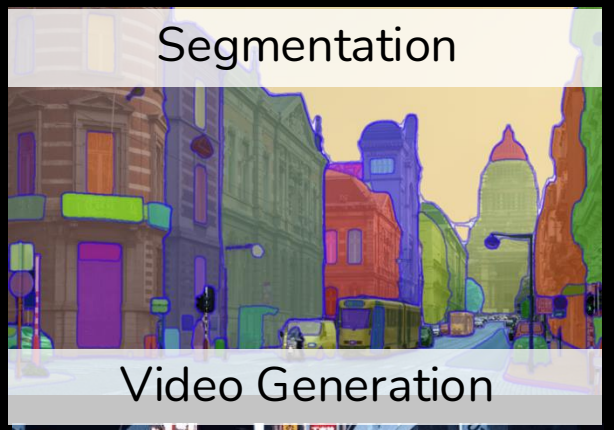
Foundation Models

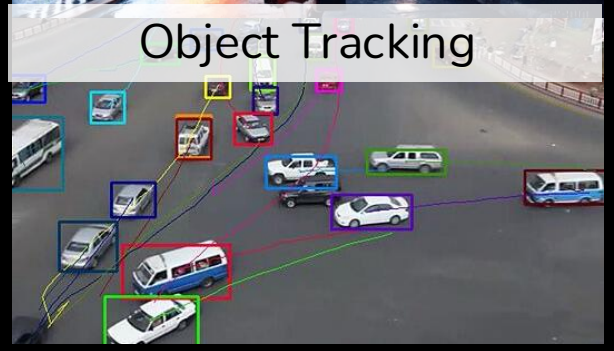
Internet-Scale Datasets

AI Foundation Model

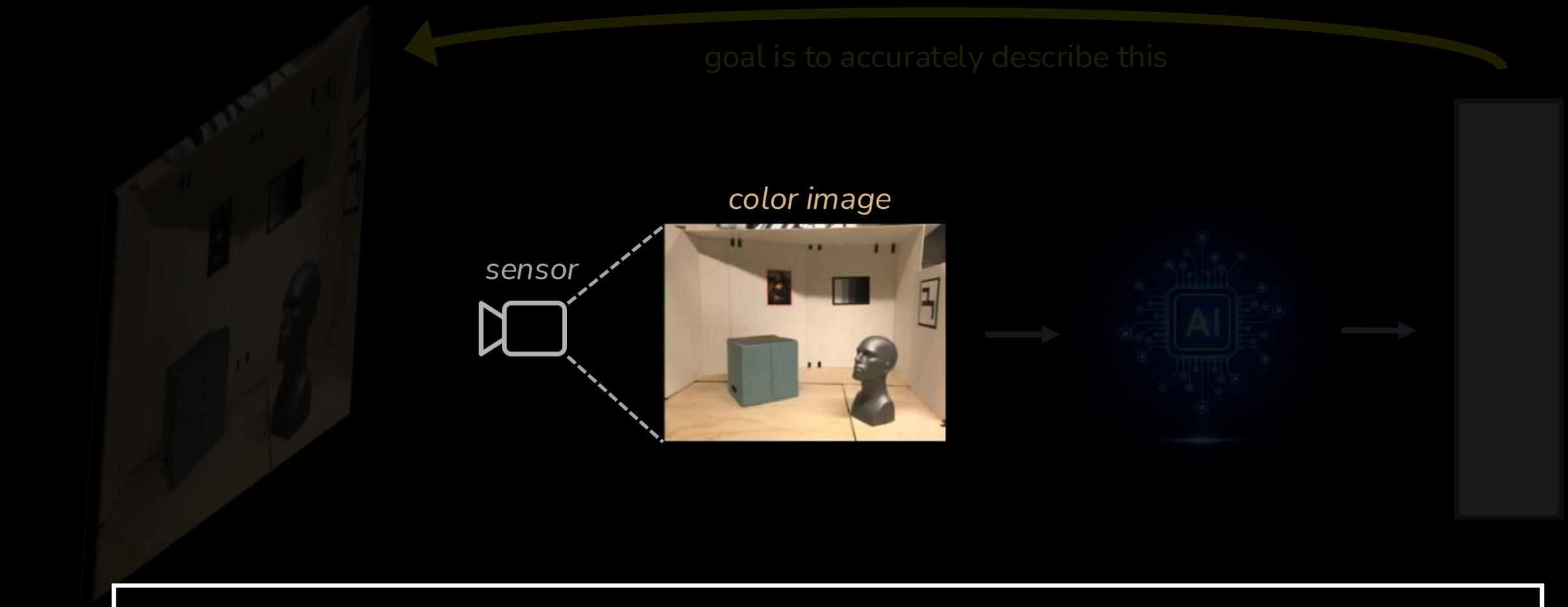


Downstream Applications





How We Train AI Models



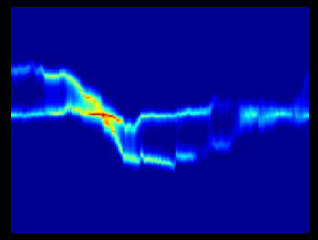
The sensor imposes a fundamental limitation on what an AI model can learn.

Solution: Use multiple modalities!

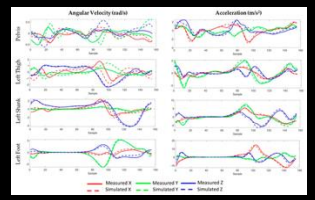
goal is to accurately describe this

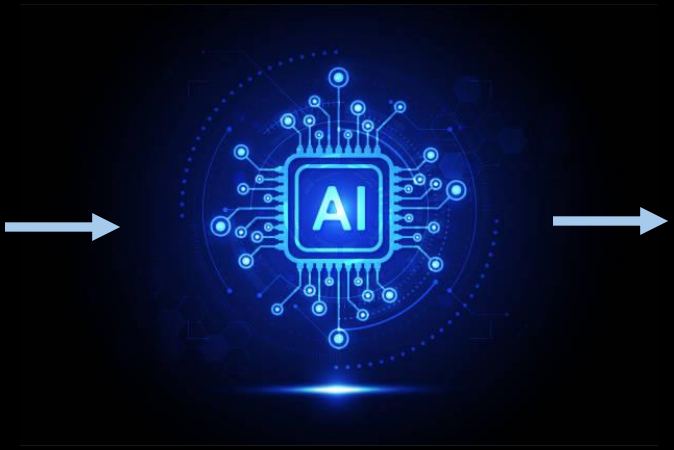
camera

WiFI



IMU





Latent Unobservable World

Many Projections of the World

AI Model

Multi-Modal Feature State

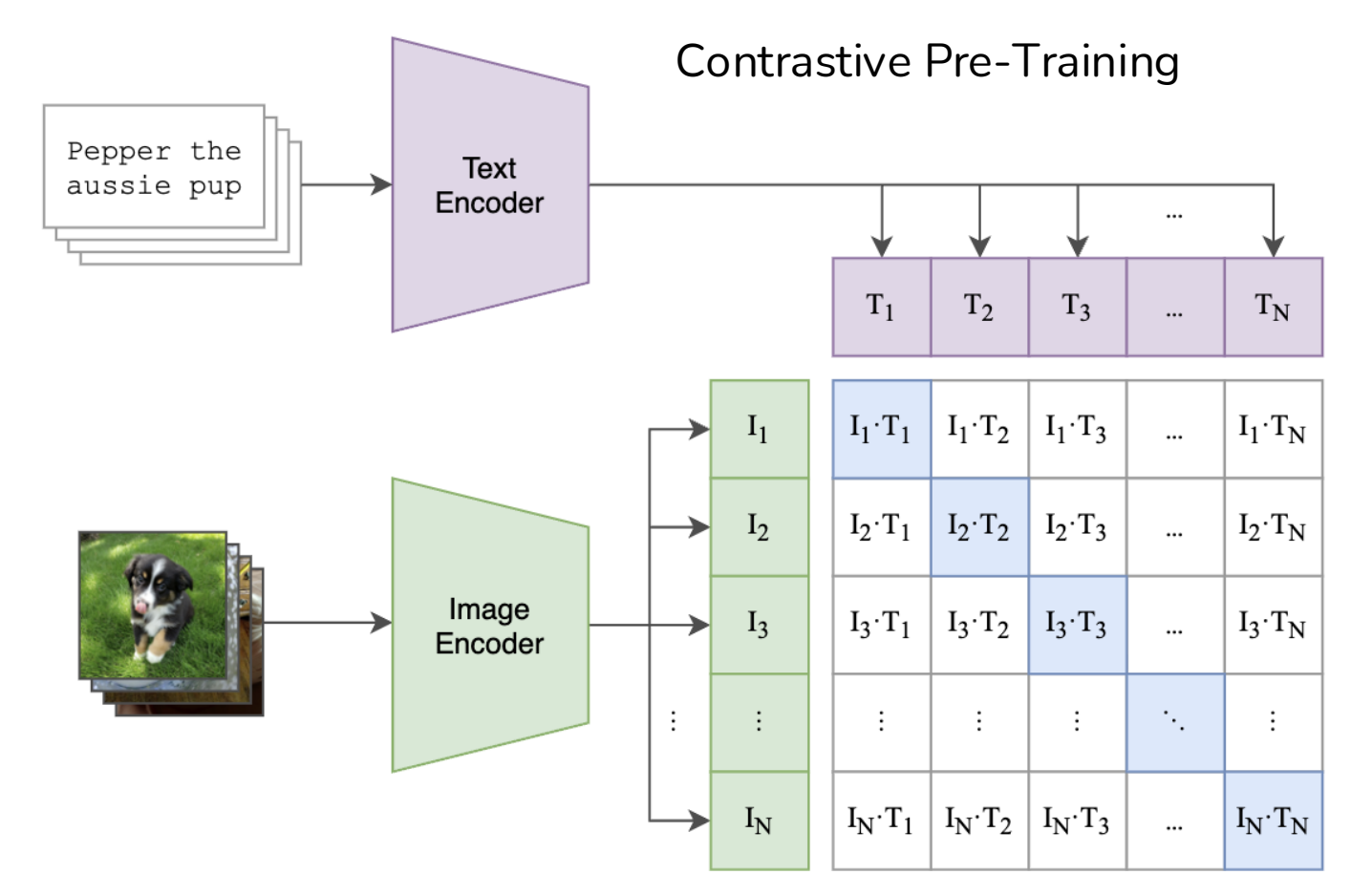
Why use multiple modalities?

- Unique information: different information in different modalities
- Robustness: shared information enforces consistency
- Flexibility: use any combination of modalities at inference

Why use machine learning?

- General representations: no need to handcraft features
- Modeling: neural networks are capable of modeling complex features
- Zero-shot transfer: generalizing to new tasks w/o extra labeled data

Multi-Modal Representation Alignment



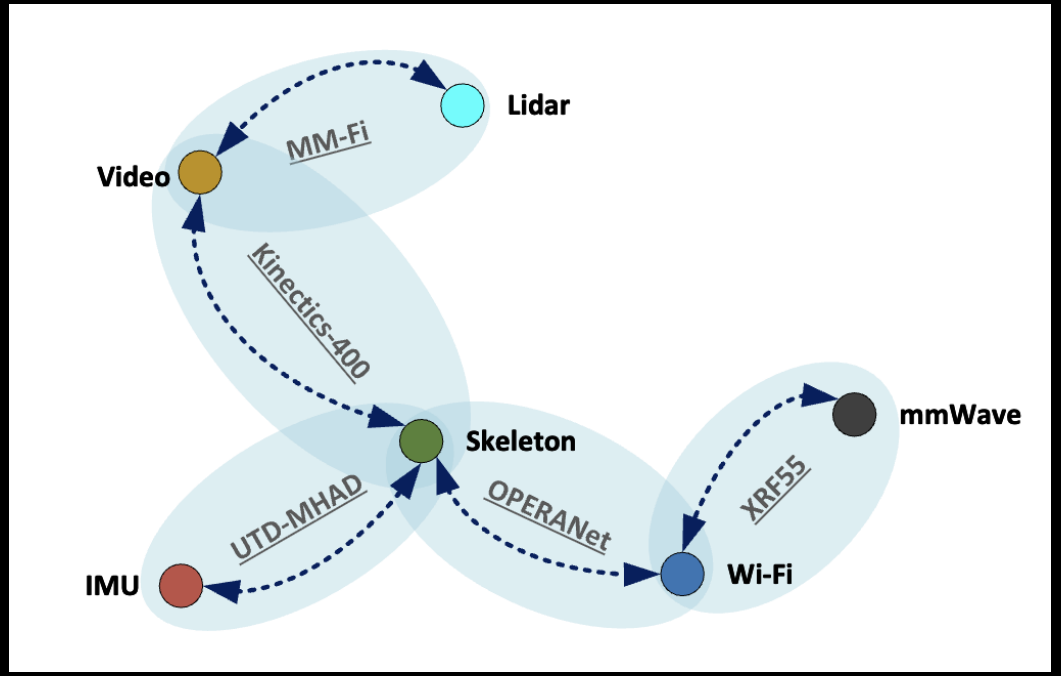

```
# image_encoder - ResNet or Vision Transformer
# text_encoder  - CBOW or Text Transformer
# I[n, h, w, c] - minibatch of aligned images
# T[n, l]       - minibatch of aligned texts
# W_i[d_i, d_e] - learned proj of image to embed
# W_t[d_t, d_e] - learned proj of text to embed
# t             - learned temperature parameter

# extract feature representations of each modality
I_f = image_encoder(I) #[n, d_i]
T_f = text_encoder(T)  #[n, d_t]

# joint multimodal embedding [n, d_e]
I_e = l2_normalize(np.dot(I_f, W_i), axis=1)
T_e = l2_normalize(np.dot(T_f, W_t), axis=1)

# scaled pairwise cosine similarities [n, n]
logits = np.dot(I_e, T_e.T) * np.exp(t)

# symmetric loss function
labels = np.arange(n)
loss_i = cross_entropy_loss(logits, labels, axis=0)
loss_t = cross_entropy_loss(logits, labels, axis=1)
loss   = (loss_i + loss_t)/2
```

Can only support two modalities and requires lot of data (~400 million paired samples).

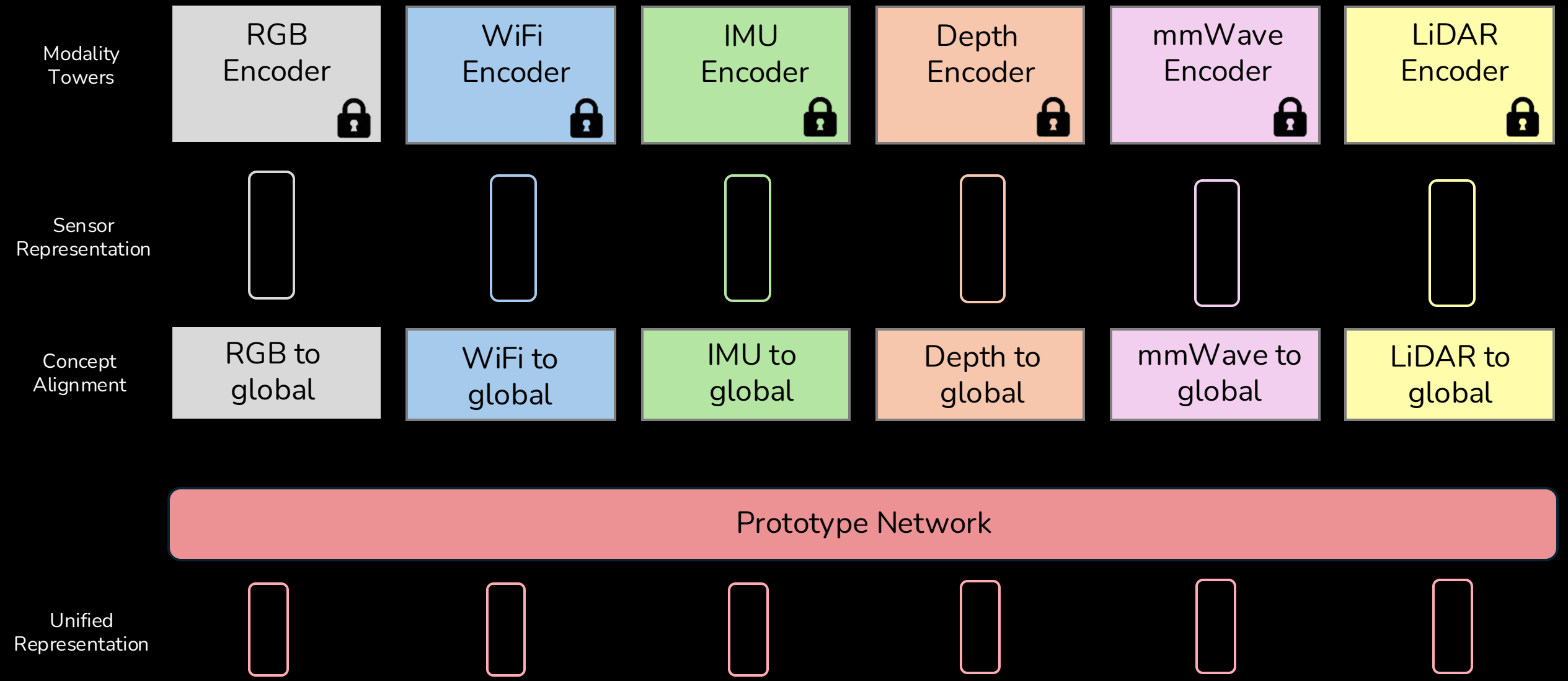
Radford et al, 2021

Key Observations

- Can leverage existing uni-modal encoders for the modalities of interest
  - Existing encoders are trained on many samples already
- There are paired datasets for at least 2 modalities



Computational Approach



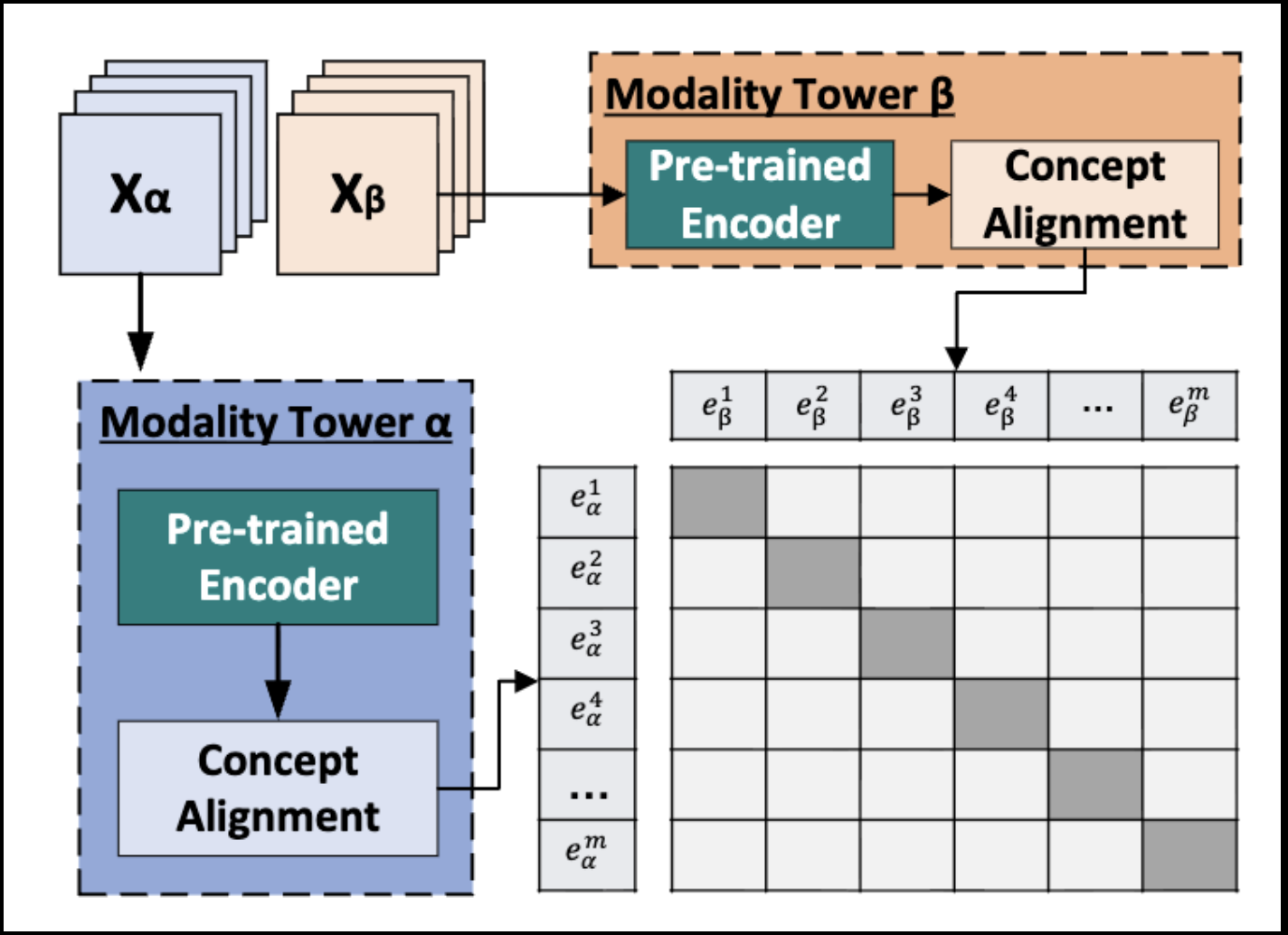
Key Considerations for Pre-Trained Encoders

- Either choose existing learned encoders or signal processing-based techniques
- Don’t want to choose models that will suffer from domain shifts (e.g. LiDAR trained on AV datasets transferred to indoor settings)
- Can use multiple encoders for a single modality to improve robustness for modalities with lot of noise
  - These representations would be aligned using contrastive learning

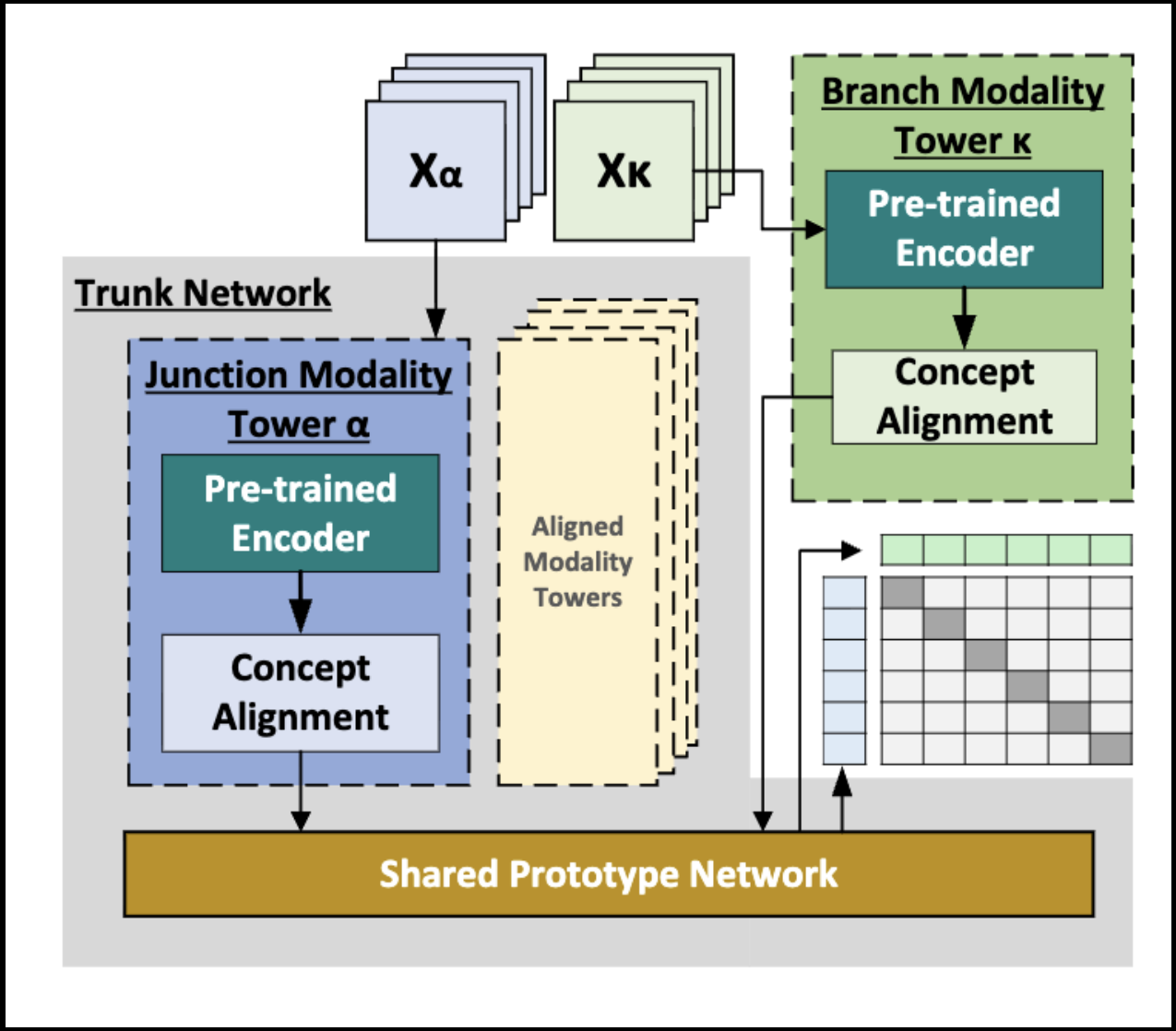
Basic Approach

- Align modality 1 and modality 2
- Once aligned, align modality 3 to modality 1 (while preserving unified representation)
- Repeat until all modalities are included

Aligning First Two Networks (Trunk Network)



Network Growth (Aligning Next Modalities)



Jointly update prototype network and concept alignment

This helps prevent catastrophic forgetting by maintaining a shared set of weights across modalities

Adaptive Training Strategy

What will happen to the representations if an added modality is more noisy/unreliable?

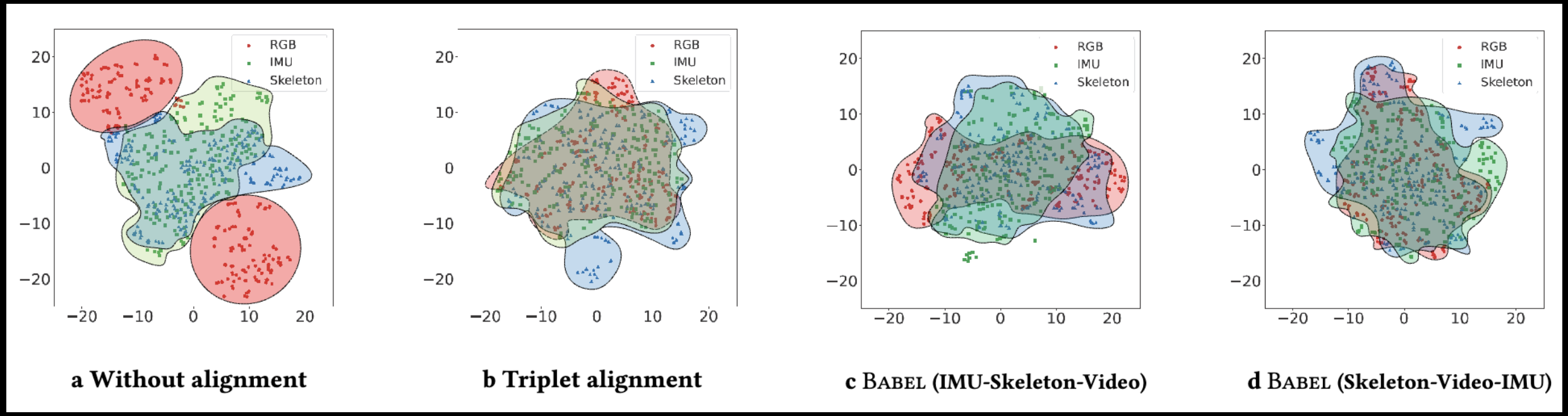
$$L_{\alpha\beta}^{M} = \frac{w_{\alpha \leftarrow \beta} \cdot L_{\alpha \leftarrow \beta}^{M} + w_{\beta \leftarrow \alpha} \cdot L_{\beta \leftarrow \alpha}^{M}}{2}$$

The weights control how much you should force one modality to be like the other or vice-versa.

$$w_{\alpha \leftarrow \beta}^{M} = \frac{1}{\| \nabla_{\alpha \leftarrow \beta}^{M}(\Gamma_{\alpha}, \Gamma_{\beta}) \|},$$

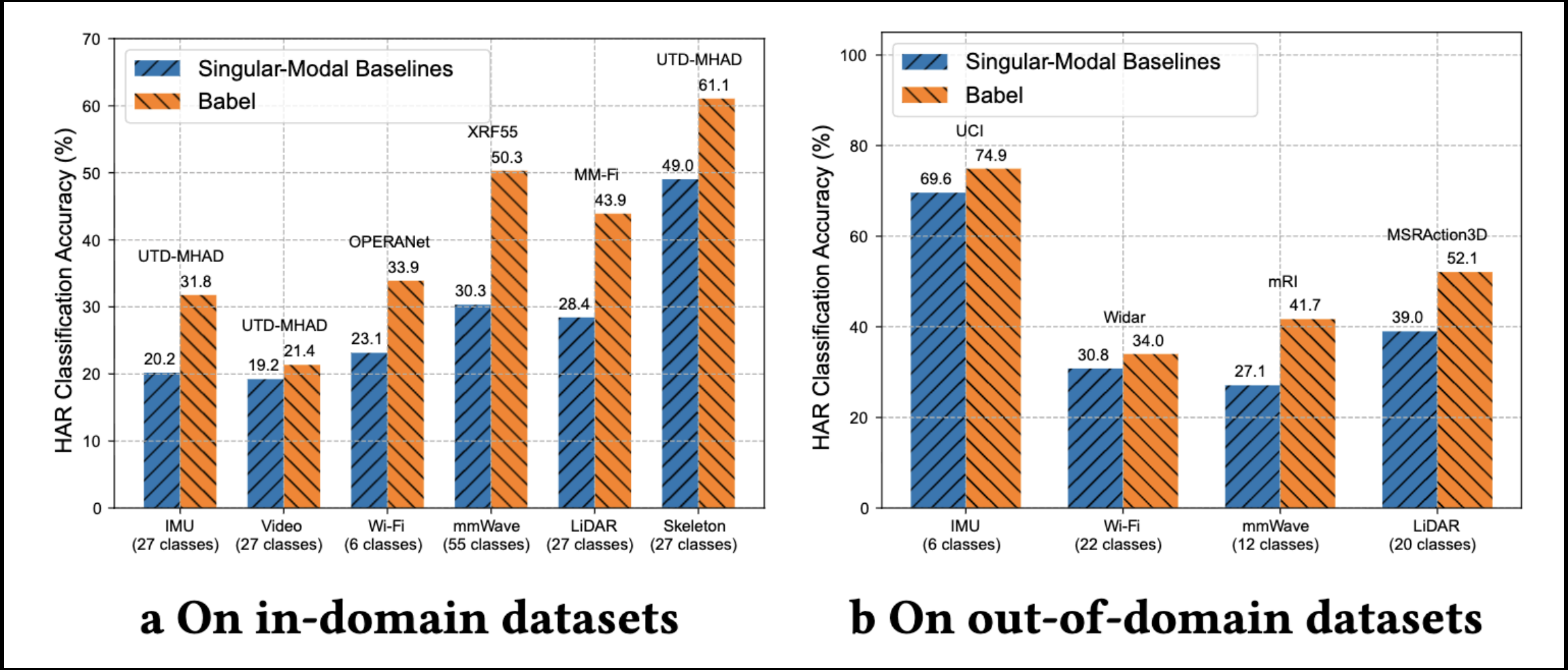
Weight is inversely proportional to “reliability” of modality

Network Growth vs. Triple Alignment



a Without alignment **b Triplet alignment** **c BABEL (IMU-Skeleton-Video)** **d BABEL (Skeleton-Video-IMU)**

Evaluation: Human Activity Recognition



Results obtained using one-shot learning.

Effect of Ordering on Performance

Heuristics	IMU UTD-MHAD [9]	Skeleton UTD-MHAD	Video UTD-MHAD	Wi-Fi OPERANet [4]	mmWave XRF55 [51]	LiDAR MM-Fi [64]
Random	31.77%	61.06%	21.35%	33.89%	50.30%	43.91%
Robustness	29.33%	60.58%	20.83%	35.31%	52.16%	44.65%
Diversity	27.60%	56.25%	21.35%	35.79%	47.93%	44.21%
Amount	28.13%	59.90%	20.83%	33.85%	46.85%	47.70%

Advances over Prior Multi-Modal Techniques

Table 2: The classification accuracy of sensing fusion for BABEL and the multi-modal sensing baselines evaluated on the out-of-domain mRI[1] dataset.

	Multi-modal Baselines	Babel
Vision+IMU	89.6%	91.7%(+2.3%)
Vision+mmWave	58.4%	64.6%(+10.7%)
IMU+mmWave	75.0%	86.5%(+13.5%)
Vision+IMU+mmWave	85.4%	92.8%(+8.6%)

Multi-Modal LLM

(Upload IMU readings that describes this action)

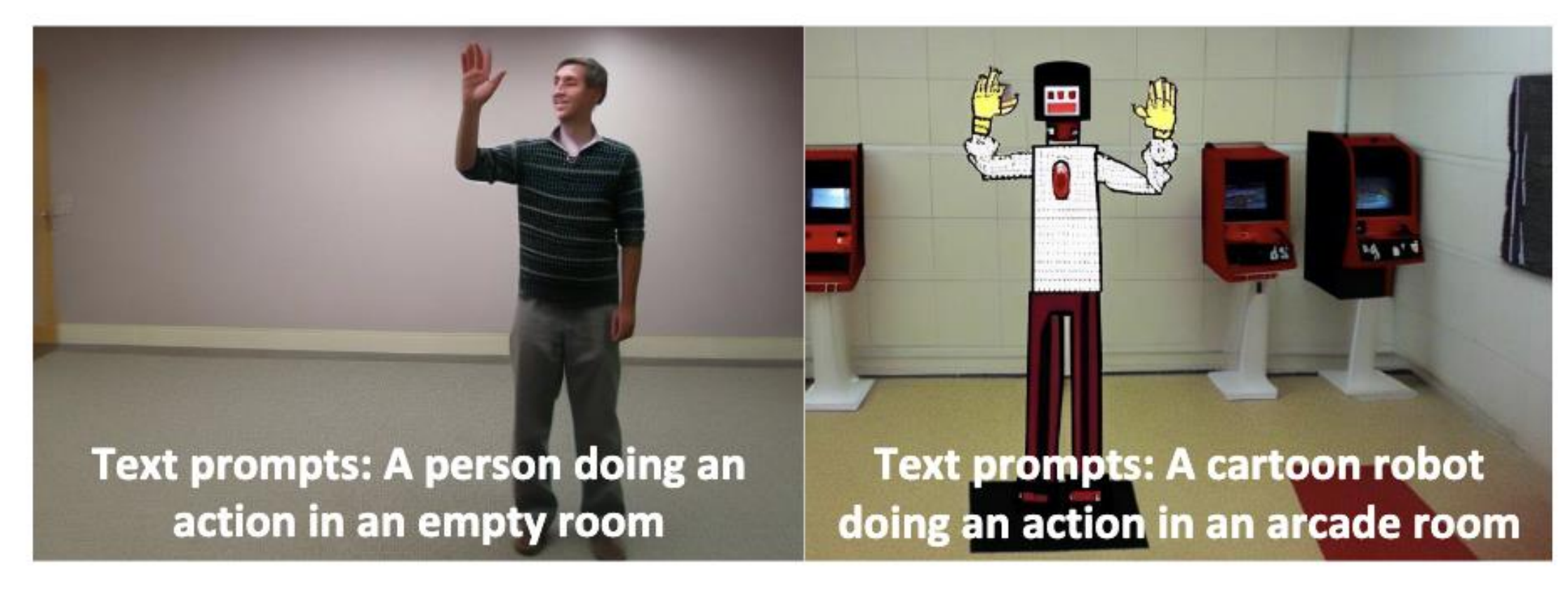
As requested, I will describe the action in IMU terms:
Body Position: The person is standing with their feed should-width part, indicating a stable base of support.
Shoulder Movement: The person's shoulders are moving in a rotational motion.
Elbow Movement: The person's arm is moving in a smooth motion, primarily in the vertical plane.
Hand Movement: The person's hands are moving in a rapid, precise motion.

Squat or right hand wave?

It suggests the person is performing a 'right hand wave'.

Reference Video

Application: Cross-Modal Rendering



IMU signal is input, and environment is specified via text.

Limitations

- Representation alignment is NOT always equal to better performance
  - Is alignment the best way to train these models?
  - Evaluated task requires only semantic understanding; what if you want spatial understanding of images?