Efficient Many-Function Video ML at the Edge

Chris Rowen
Will Reed

Collaboration AI
Cisco Systems

CISCO

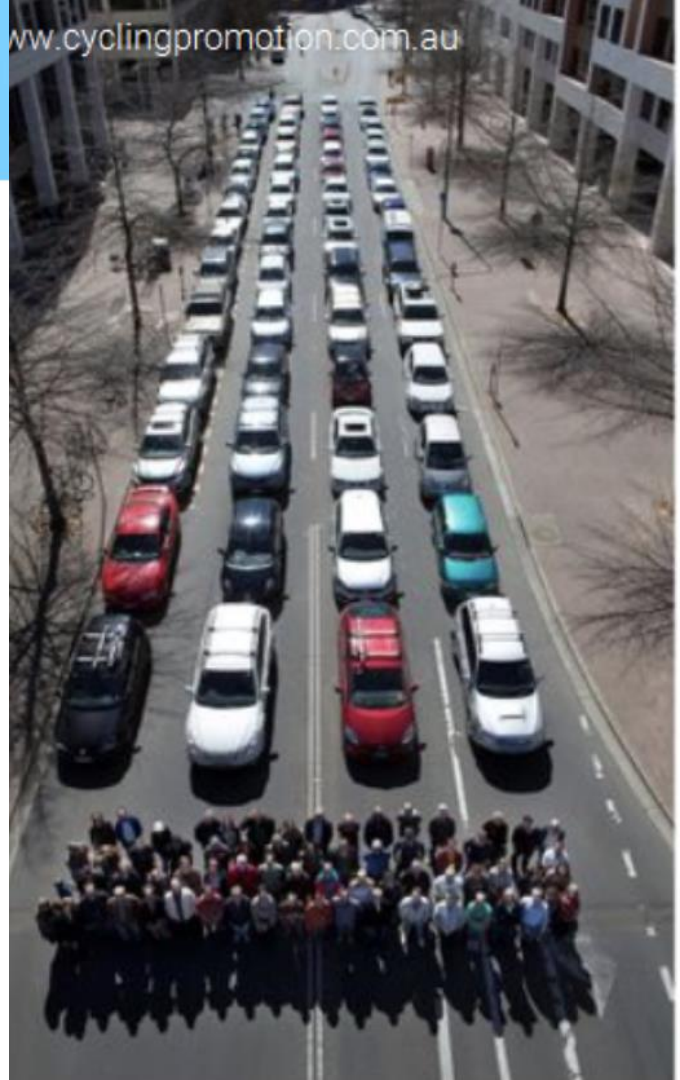
The Problem

- Many ML tasks
- Limited capacity:
  - Compute
  - Memory
  - Development time

cisco

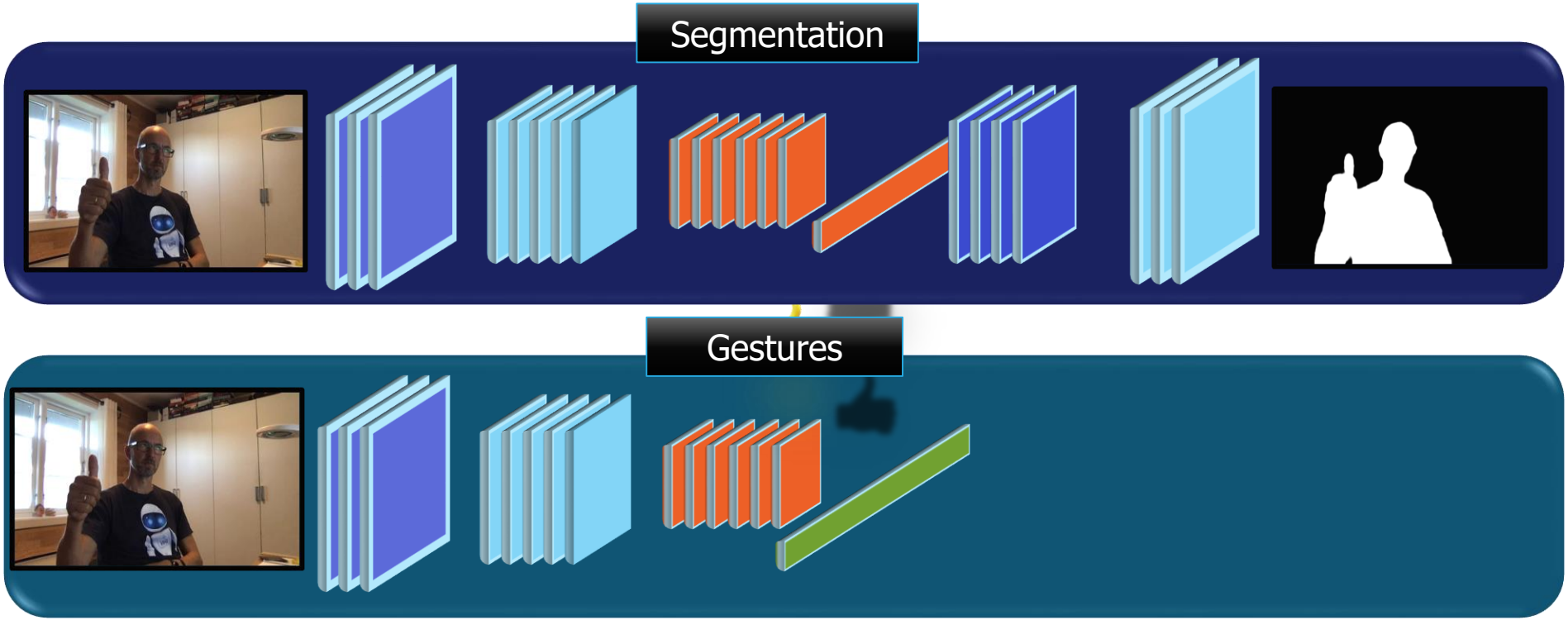
Our Solution

- Many tasks, 1 model
- Share common paths
  - Architecture
  - Data
  - Testing
  - Deployment

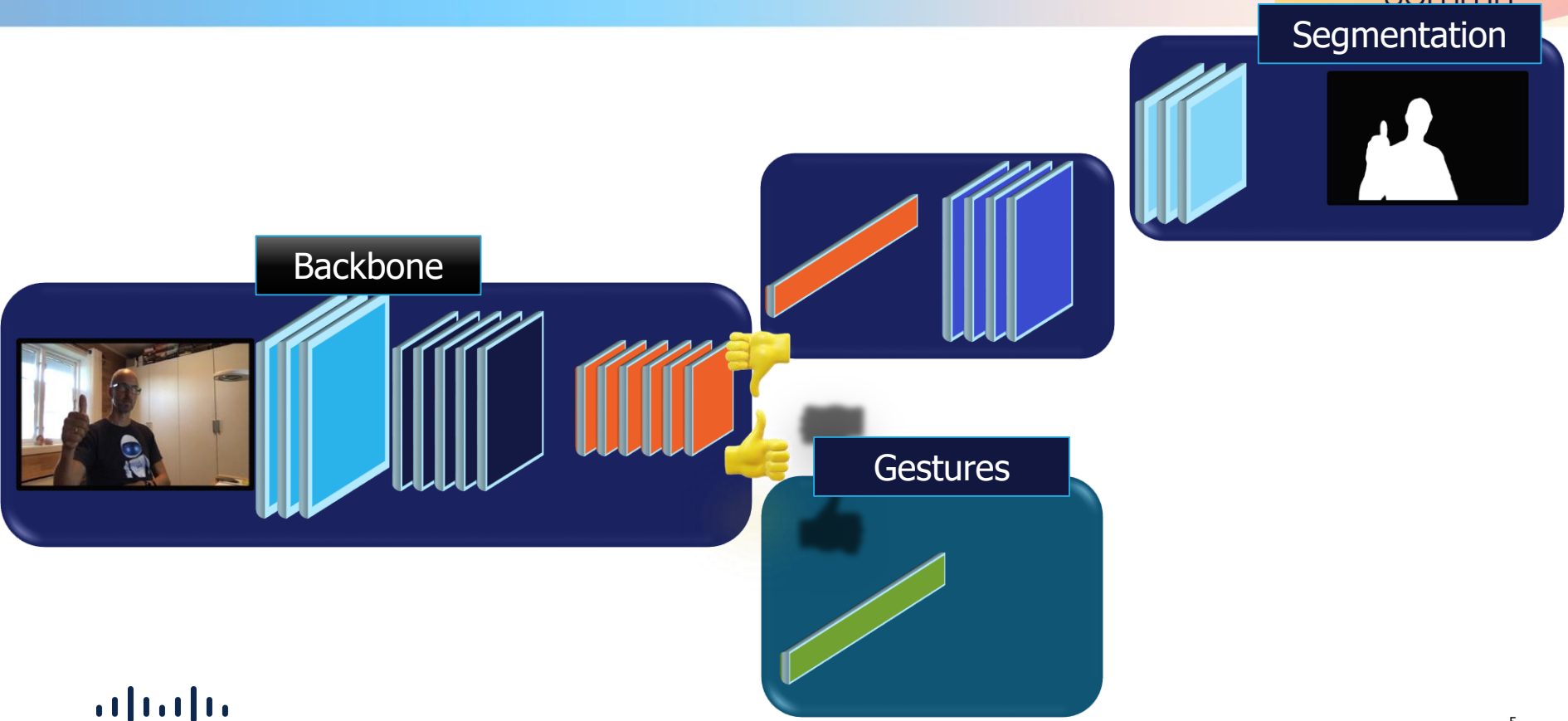


cisco

Before: 2 Tasks, 2 Models



After: 2 Tasks, 1 Model



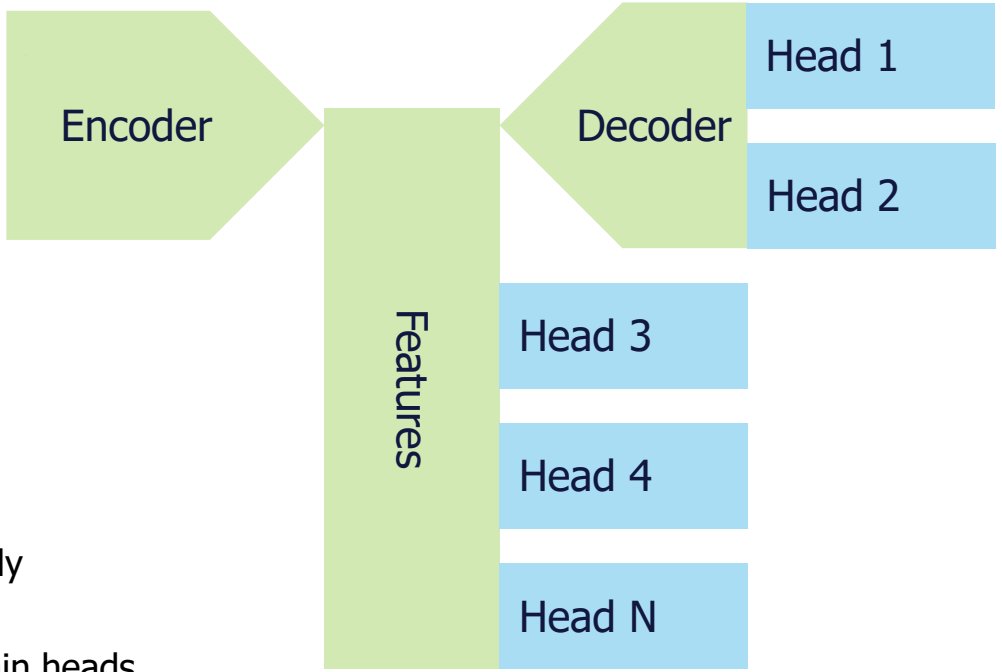
Model Size Comparison: 2 functions

	# Encoder Parameters	# Models	# Head Parameters	TOTAL
Before:	2.4M	2	300k 50k	5.15M
After:	2.4M	1	300k 50k	2.75M

Architecture

video frames

- Configurable input size
- Swappable encoders
  - 200 K, 420 K and 2.4 M param options
- Configurable training
  - Train encoder, features and decoder jointly
  - Freeze encoder and train heads
  - Train encoder, lower learning rate and train heads



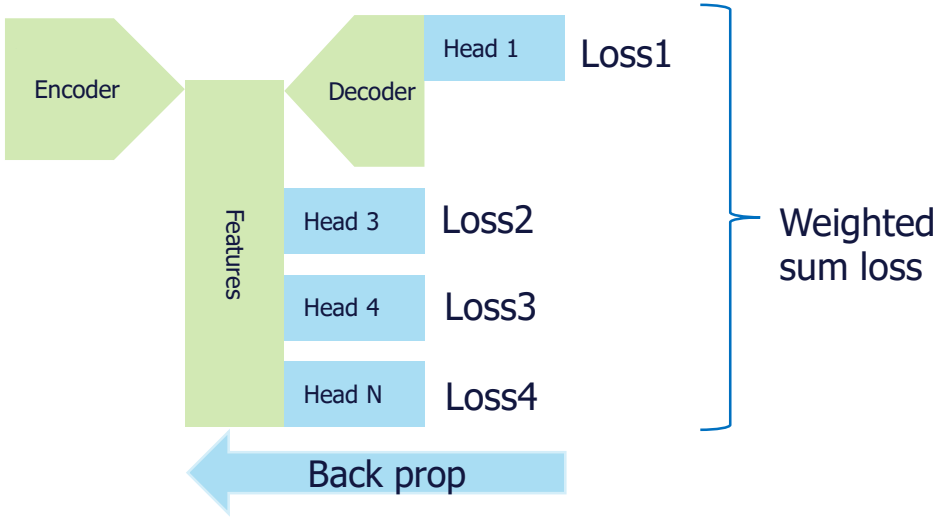
Data Strategy

- Data efficient architecture
  - Tasks can learn from each other's data through generalization in the encoder
  - Similar to how models benefit from pre-training
- Implicit regularization
  - Multiple tasks discourage the model from overfitting on any one task
- Benefit from diversity of data in related tasks
- Only need to label data for the task you care about
  - Quick and cheap to add a new task

Training

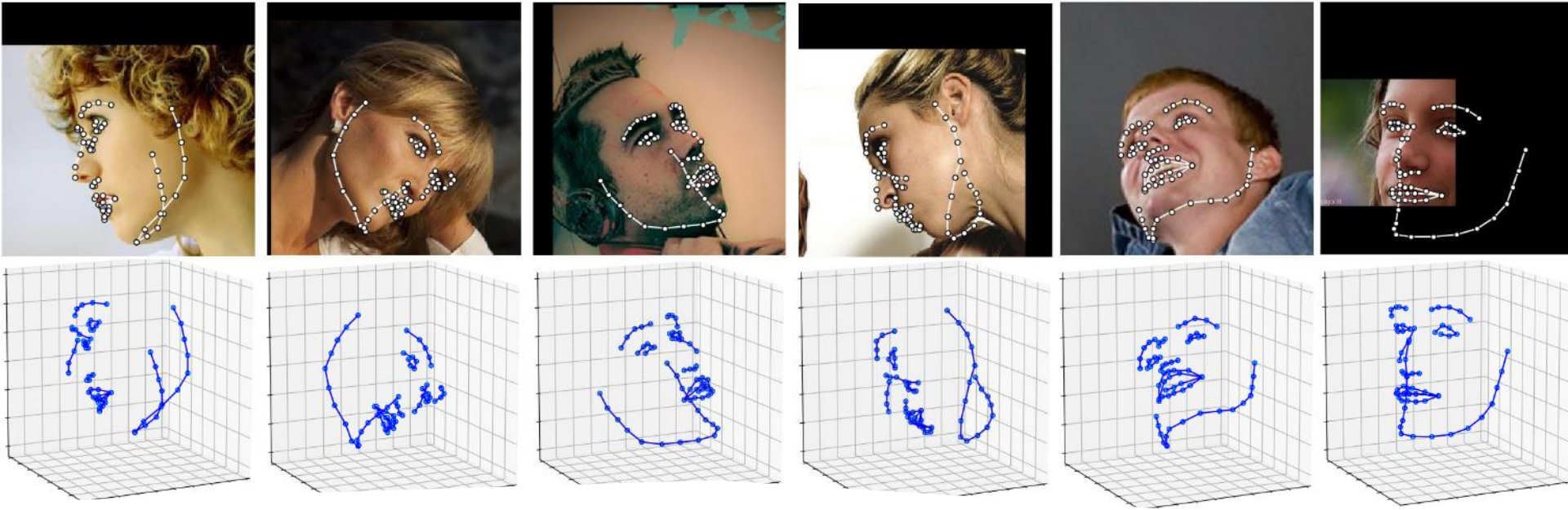
Inputs

Function Labels



- Skip losses for unlabeled functions

Adding a New Task: Face Landmarks Detection



SSLD: Single Shot Landmark Detection

- Perform face and keypoints detection in one pass
- Based on YOLO v2 (transitioning to v3 soon)
- Remove classification loss
- Add landmark localization loss
- Computation is bounded

$$\lambda_{\text{coord}} \sum_{i=0}^{M\times N}\sum_{j=0}^{A} 1_{ij}^{\text{obj}} \left[\left(x_i - \hat{x}_i\right)^2 + \left(y_i - \hat{y}_i\right)^2 \right]$$ *face center*

$$+\lambda_{\text{coord}} \sum_{i=0}^{M\times N}\sum_{j=0}^{A} 1_{ij}^{\text{obj}} \left[\left(\sqrt{w_i} - \sqrt{\hat{w}_i}\right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i}\right)^2 \right]$$ *face box*

$$+ \sum_{i=0}^{M\times N}\sum_{j=0}^{A} 1_{ij}^{\text{obj}} \left(C_i - \hat{C}_i\right)^2$$ *face classification (presence)*

$$+\lambda_{\text{noobj}} \sum_{i=0}^{M\times N}\sum_{j=0}^{A} 1_{ij}^{\text{noobj}} \left(C_i - \hat{C}_i\right)^2$$

$$+ \sum_{i=0}^{M\times N}\sum_{j=0}^{A} 1_{ij}^{\text{obj}} \sum_{c\in\text{classes}} \left(p_i(c) - \hat{p}_i(c)\right)^2$$ *classification confidence*

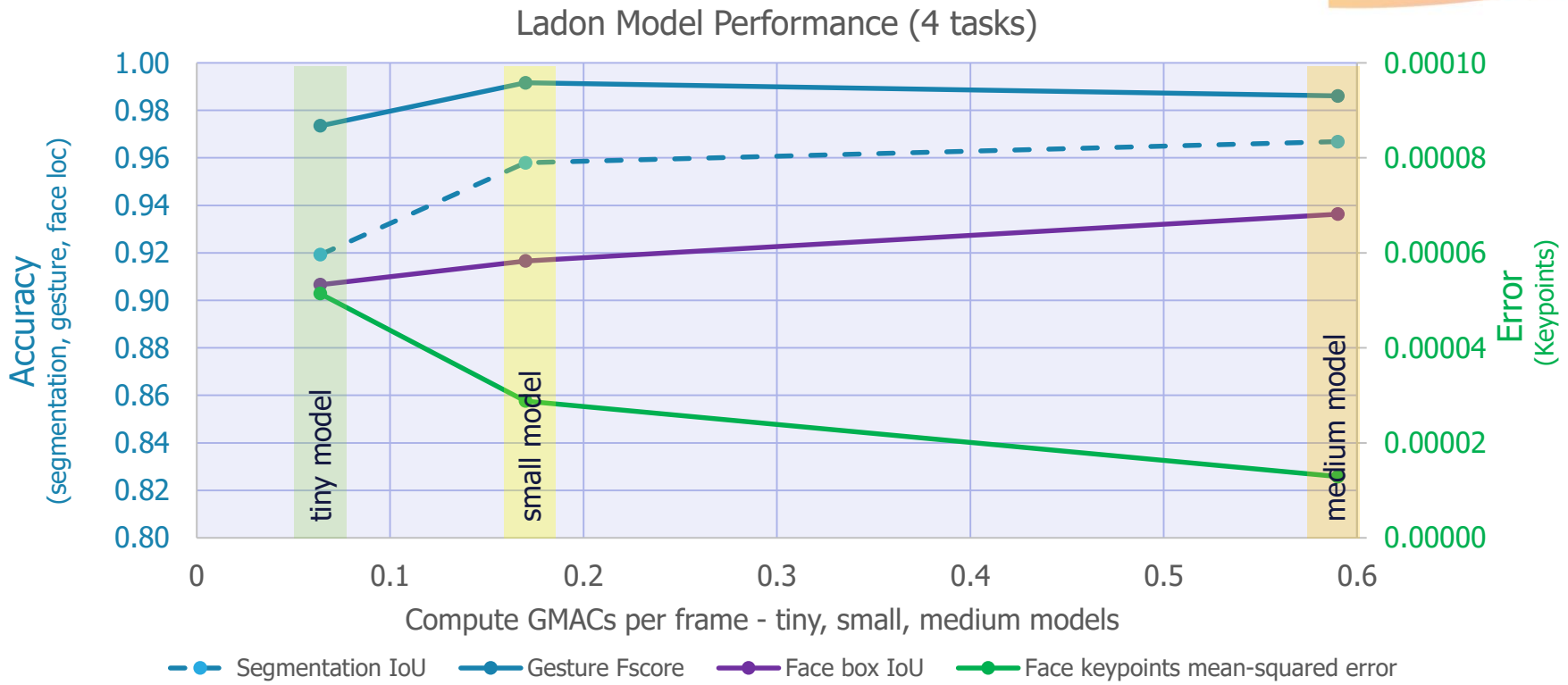
$$\lambda_{coord} \sum_{i=0}^{M\times N}\sum_{j=0}^{A} 1_{ij}^{obj} \left[\left(x_i - \hat{x}_i\right)^2 + \left(y_i - \hat{y}_i\right)^2 \right]$$ *landmark locations*

Model Size Comparison: 4 functions

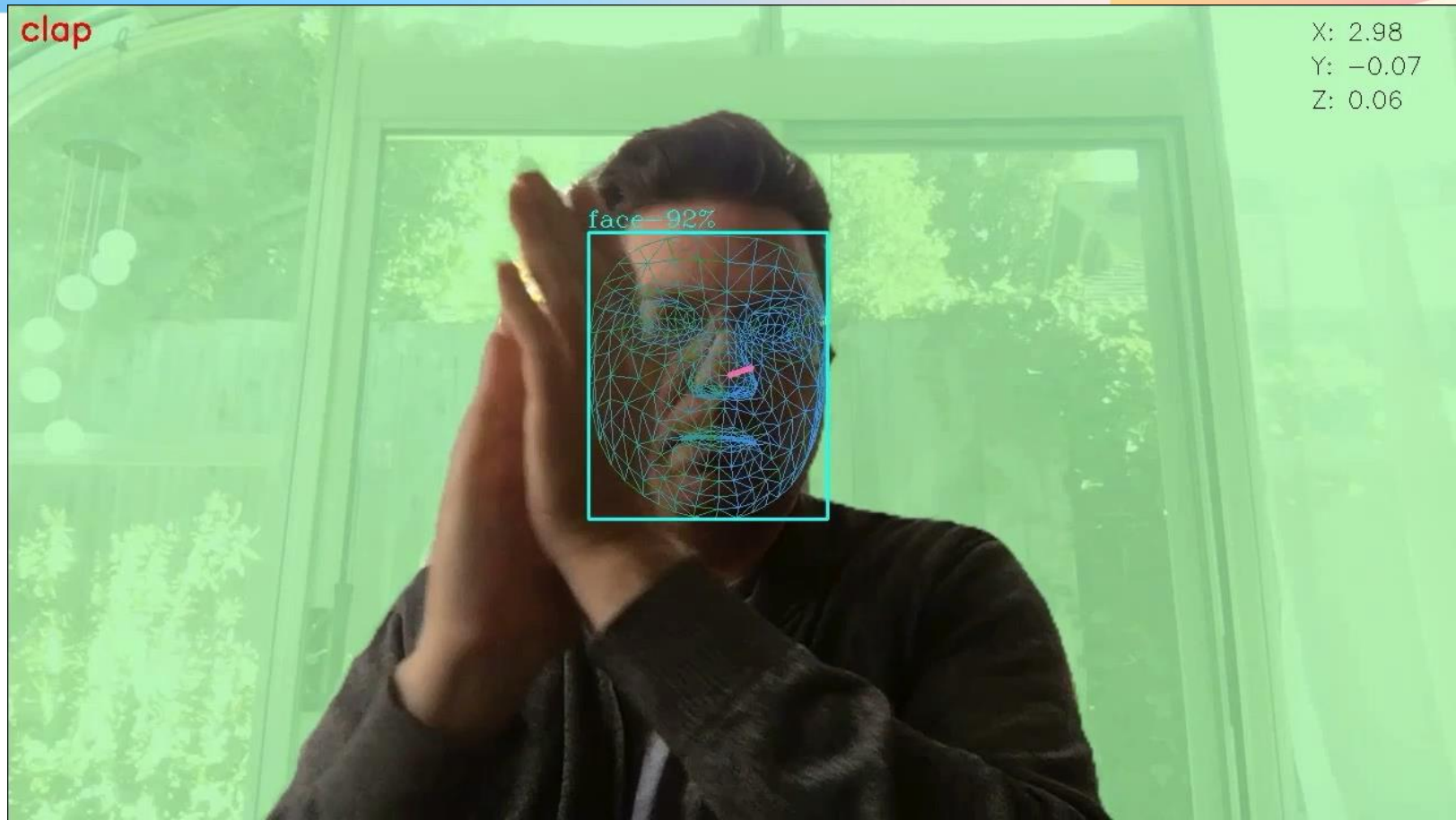
segmentation + gesture + face location+ face landmarks

	# Encoder Parameters	# Models	# Head Parameters	TOTAL
Separate:	2.4M	4	300k (segment) 50k (gesture) 50k (landmarks) 30k (face loc)	11.2M
Unified:	2.4M	1	300k (segment) 50K (gesture) 50K (landmark+face loc)	2.8M

Multi-Function Model Performance

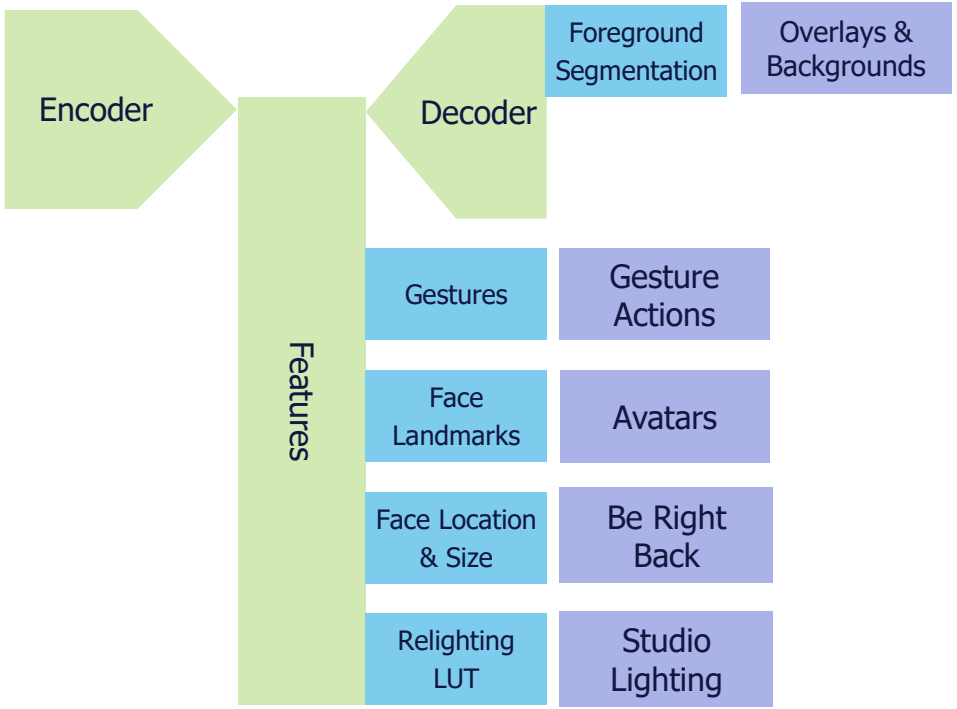


Examples



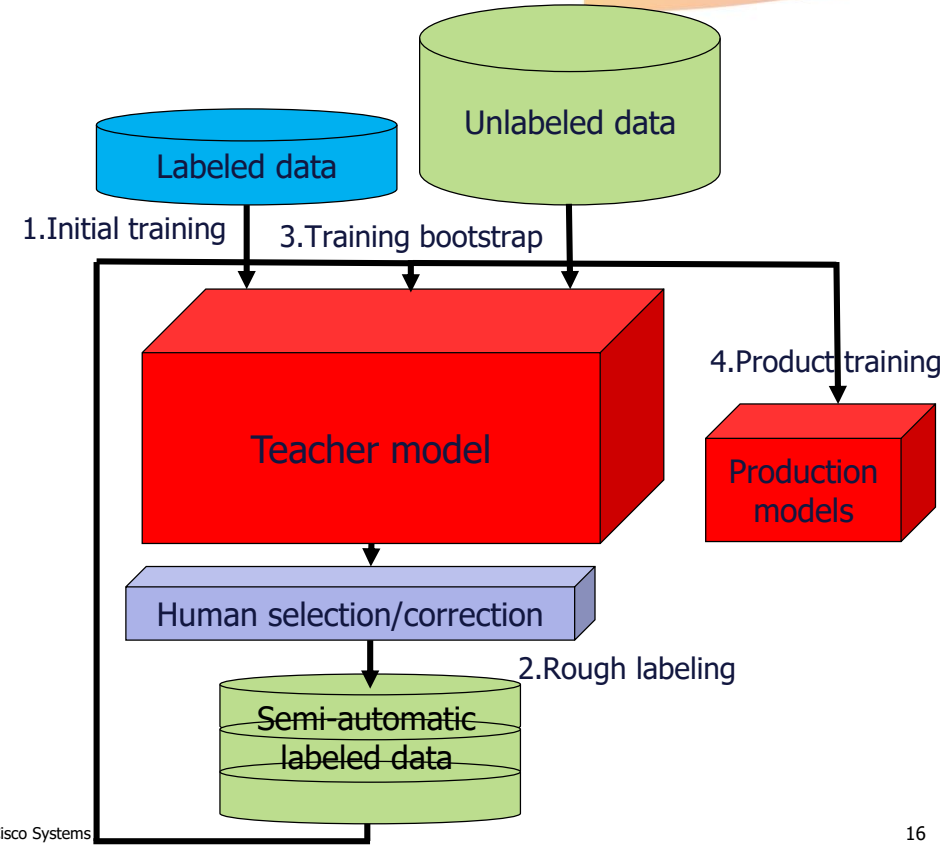
Optimized Portable Edge Implementation

- Cross-platform
  - C++: Supports Windows, Mac, Linux, iOS, Android
  - Javascript: Recent versions of major browsers
- Common ML framework
  - ONNX, CoreML, OpenVINO
  - CPU and GPU mode
  - Consistent results
  - Less conversion hell
- High and low level API's
  - Get high level predictions like a fully segmented and blurred output, 2 and 3D filter effects
  - Also low level access to segmentation masks, landmark coordinates, etc…



Training Label & Data Synthesis

- Two powerful methods
  - Teachers:
    - Build big teacher model from limited hand-labeled data
    - use it to bootstrap big data set
    - Train range of small models
  - Synthetic Scenes:
    - Diverse
      - 3D environments
      - faces
      - poses
      - lighting
    - Programmatically animate 3D avatars



Lessons Learned

1. Richer applications ➔ explosion of video ML needs ➔ compute crisis?
2. Multi-headed vision model is a form of ”foundation model” like GPT-4 ➔ robustness through task diversity
3. Smart algorithmic labeling can replace much hand labeling
4. Estimating algorithm FLOPS is an imperfect predictor of implementation throughput – not every layer is a convolution
5. Implementing N functions together complicates loss function design and training
6. Raising system functionality may span many platforms ➔ portability
7. Emergence of edge CPU neural accelerators may open door to more aggressive video ML workloads, but uneven time-lines for availability
8. Conventional wisdom says diverse training tasks together often doesn’t work. Conventional wisdom is often wrong.

Resources

- Semantic Segmentation: https://www.v7labs.com/blog/semantic-segmentation-guide
- Adaptive Re-lighting: https://arxiv.org/pdf/2009.14468.pdf
- Original YOLO paper: https://arxiv.org/abs/1506.02640
- Knowledge distillation: https://www.v7labs.com/blog/knowledge-distillation-guide
- Multi-task learning: https://towardsdatascience.com/multi-task-learning-in-machine-learning-20a37c796c9c
- Dataset Distillation: https://ai.googleblog.com/2021/12/training-machine-learning-models-more.html
- Intro to self-supervised learning: https://ai.facebook.com/blog/self-supervised-learning-the-dark-matter-of-intelligence/