# Challenges and Approaches for Cascaded DNNs: A Case Study of Face Detection for Face Verification

Dr. Ana Salazar
September 2020

# Overview

- Imagination Technologies
- Introduction to the problem
- Accuracy metrics for quantized networks
- Impact of quantisation on a one stage task: Face Detection
- Impact of quantisation on a two stages task: Face Recognition
- Experiments
- Results

# Imagination Technologies

- World leading technologies in GPU, AI, Wireless Connectivity IP and more
- 900 employees worldwide – 80% engineers
- An original IP portfolio with a significant, long-present & long-term, patent portfolio underpinning it
- Domain expertise in GPU, AI, CPU & Connectivity
- Targeting the fastest growing market segments including Mobile, Automotive, AIoT, Compute, Gaming, Consumer
- Customers include MediaTek, Rockchip, UNISOC, TI, Renesas, Socionext, and more

# Introduction to the problem

Imagination

# Embedded computer systems

- Embedded computer systems come with specific restrictions with respect to the intended application (GPU, NN accelerator, ISP), they have restricted power, system resources and features.
- In the case of NN, quantisation is a common way to  accelerate NNs; with the expectation of a minimal impact on accuracy even for 8 bit/4 bit quantisation. In this work we aim for 16 bit and 8 bit quantisation.
- We worked on a system for face recognition: Face detection followed by face verification.
- We explored the impact of quantisation on face detection first; then a face verification NN on 32fp, 16bit and 8 bit was used to verify the identity of the person

# Accuracy Metrics

Often, the accuracy of a quantised NN (e.g. 16/8 bit) is assessed using the floating point (32 bits) accuracy as reference.

This is done using the accuracy metric for the specific task:

- Classification: Top1/Top5
- Object Detection: mAP (Mean Average Precision)
- Object Detection: LAMR (Logarithm Average Miss Rate )
- Semantic Segmentation: mIOU (Mean Intersection Over Union)
- Semantic Segmentation: Overall Pixel Accuracy

When reviewing the literature of the state of the art for quantisation methods, results are reported in terms of the accuracy metrics. Not very often the accuracy metric is sensitive to quantization loss.

# Accuracy Metrics

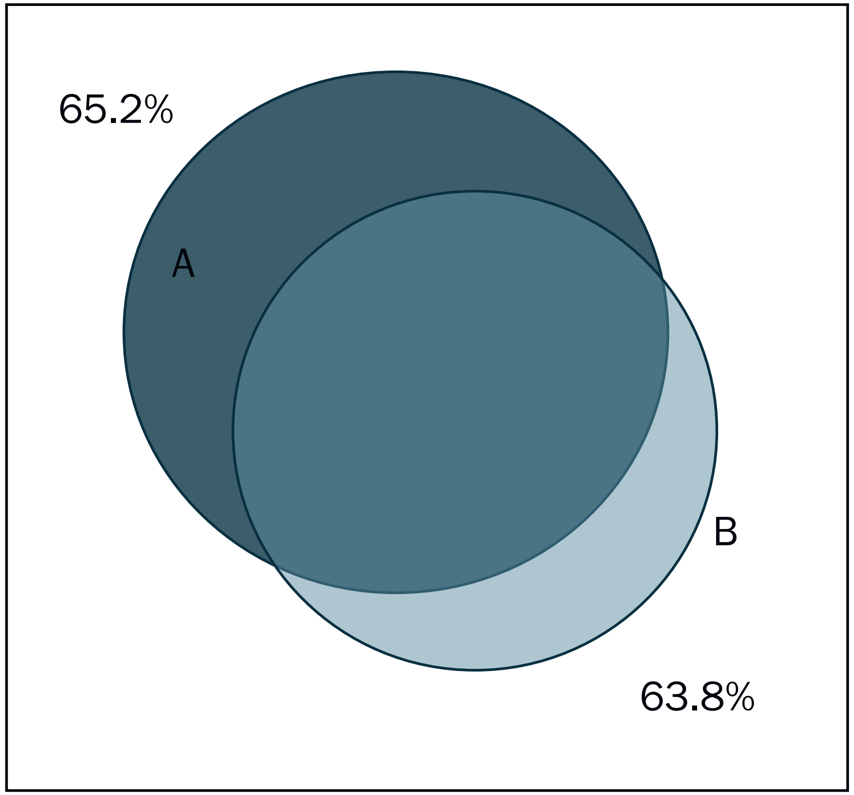
Image Classification (GoogleNet )

Dataset: ImageNet 1000 images

32 fp   Top1 65.2% -> set A

8 bits Top1 63.8% -> set B

- Top1 accuracy between 32 fp and 8 bits is close (1.4% difference)
- A represents the set of images classified correctly when using 32fp
- B represents the set of images classified correctly when using 8 bits
- B is smaller than A, but B is not a subset of A. Some of the images are classified correctly by 8 bit fixed point but not by 32fp, and vice versa.
- In this example: The overlap between set A and set B is 87.1%.




Imagination

# Accuracy Metrics

## Semantic Segmentation (DeepLab V3 ResNet V1)

Input Image

mIOU 32 fp = 0.749
mIOU 8 bits = 0.746

# Accuracy Metrics

Semantic Segmentation (DeepLab V3 ResNet V1)

32fp

8 bits quantized

# Face Recognition

Face Recognition

- Face identification
- Face verification

- Face verification validates the claimed identity using a single labelled image as reference. The output is the confirmation or rejection of the identity claimed. e.g. passport picture verification.
- Face identification identifies a person shown in an unlabelled image against a database.
- The first step for face recognition is face detection. Face detection can be seen as an object detection problem with two classes: Face / Background.

# Face Recognition as a single task

| | | | | |
|---|---|---|---|---|
| Face Detection (32fp) | + | Face Verification (32fp) | = | Accuracy Results (32fp) |
| Face Detection (8 bits) + Qerror 1 | + | Face Verification (8 bits) + Qerror 2 | = | Accuracy Results (8 bits) + Qerror1 + Qerror2 |

Our method:
The result of an 8 bits quantised face detection net will be submitted as input to an 8bit quantised face verification net.

# Face Detection
# LFFD: A light and fast face detector for edge devices

LFFD considerably balances both accuracy and running efficiency.

LFFD proposes an efficient backbone with eight detection branches. The proposal is based on qualitative analysis on pairing face scales and RF (receptive field) sizes by understanding the insights of ERF (effective receptive field), The backbone only consists of common layers (conv3×3, conv1×1, ReLU and residual connection).

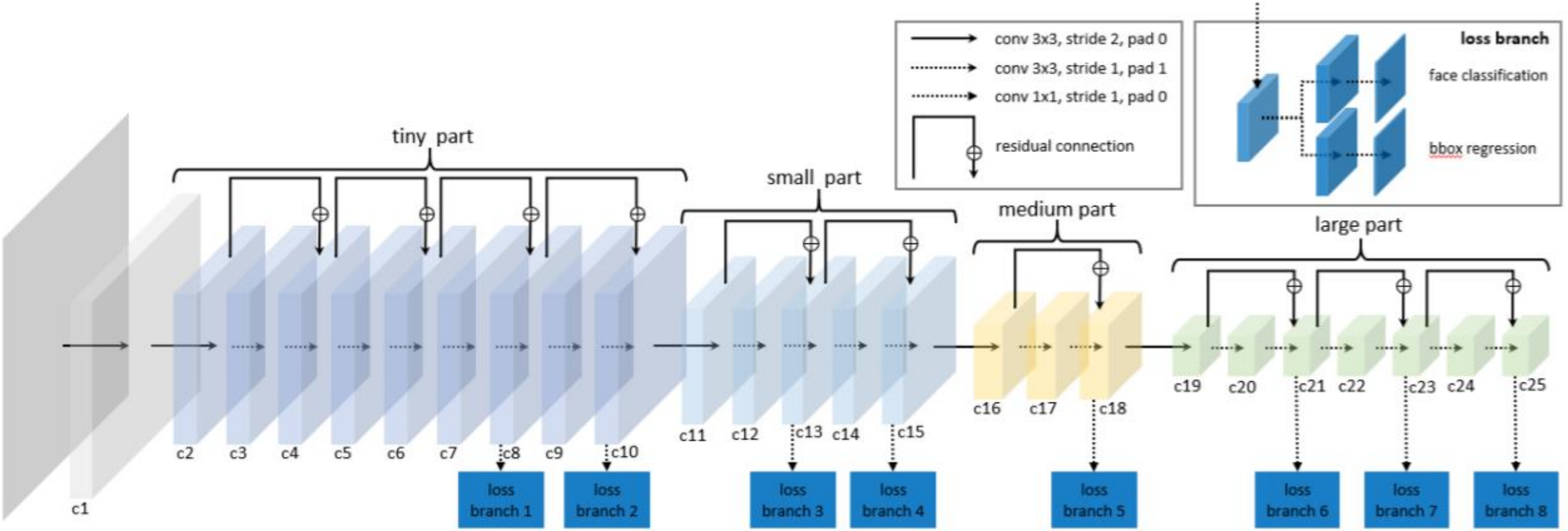
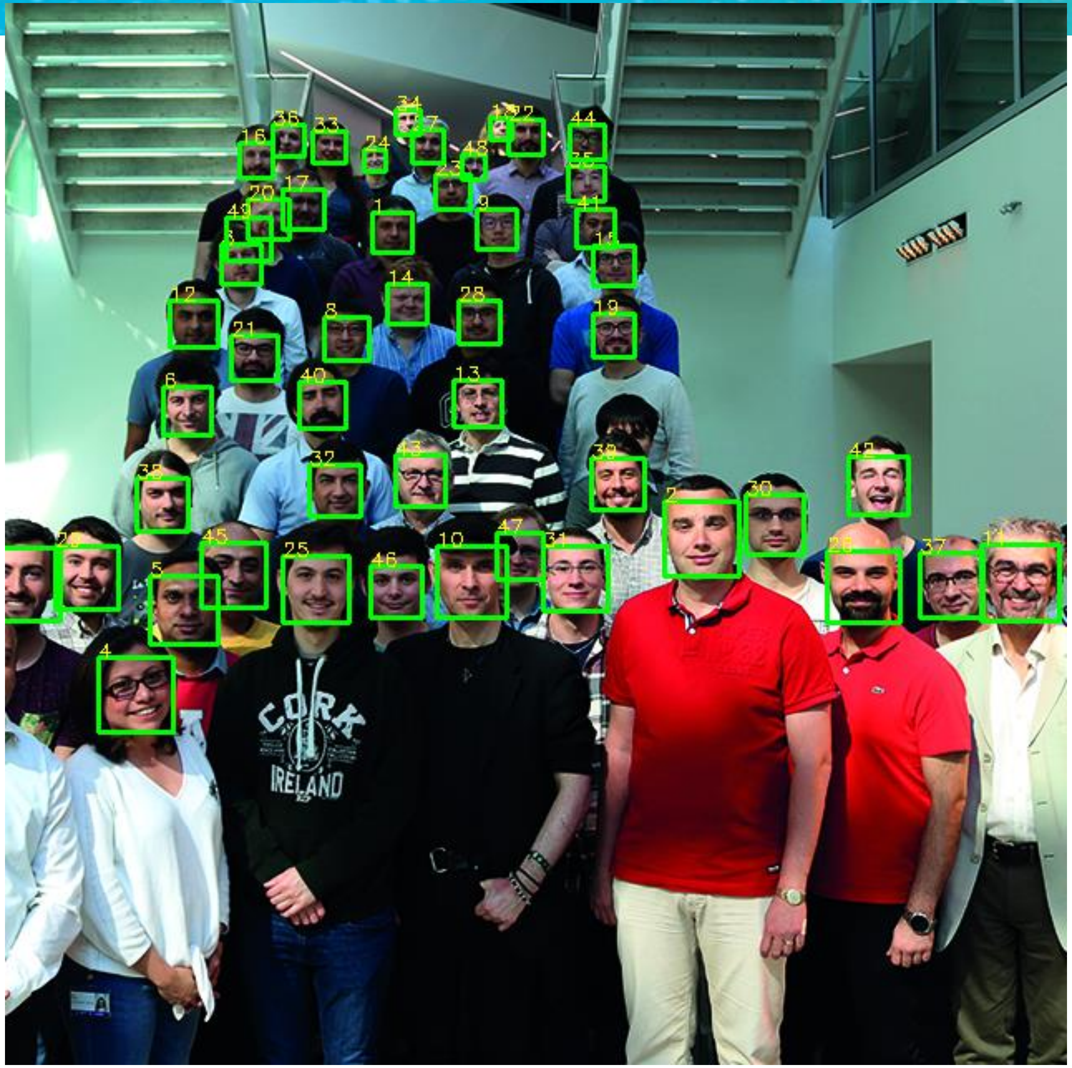


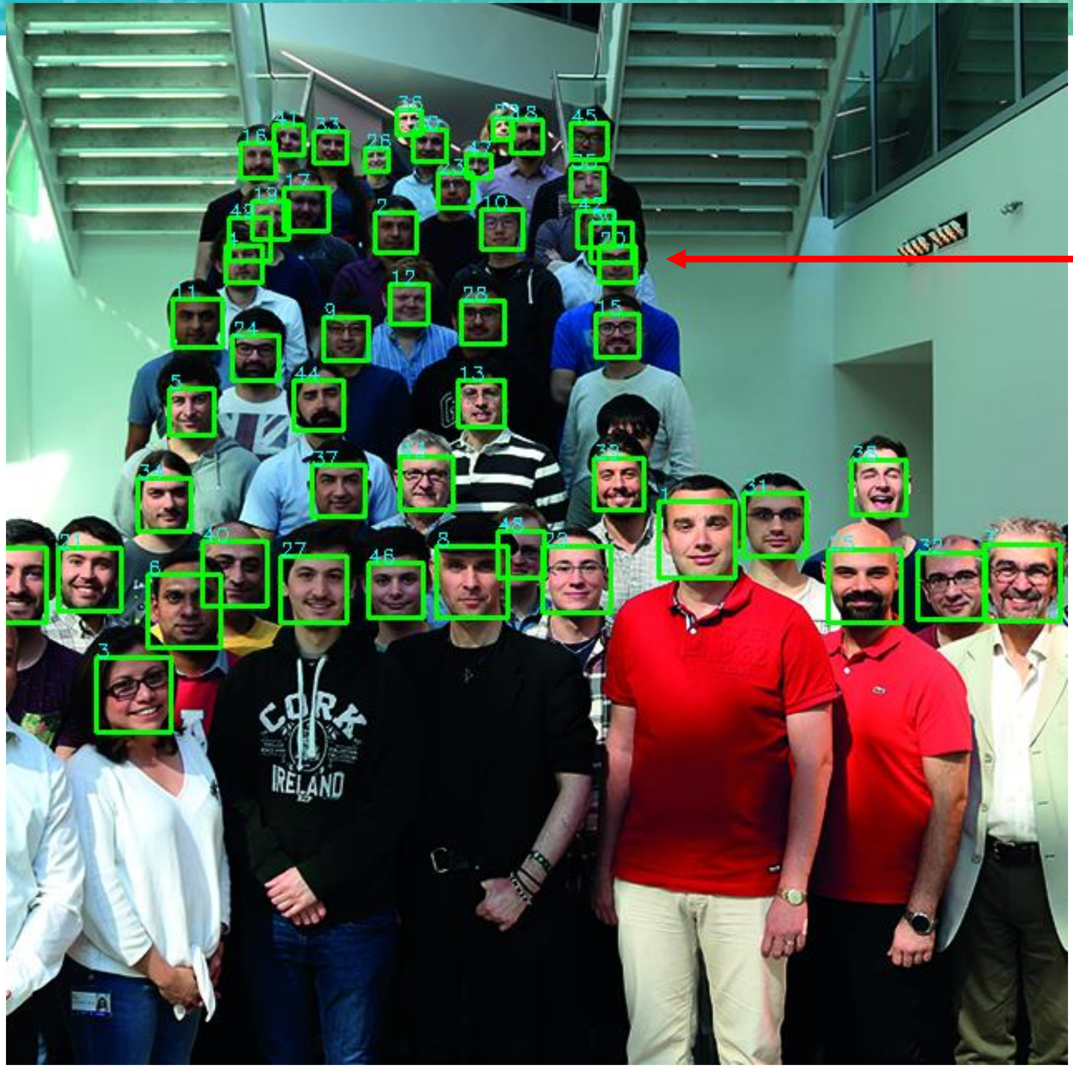

Image Credit
https://arxiv.org/pdf/1904.10633.pdf

# Face Detection

# Face Detection 32fp vs 8 bits




32fp



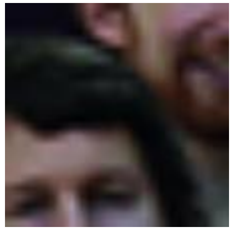

8 bits

Extra false positive

# Face detection

- LFFD f32 detections has 1 false positive
- LFFD 16 bits has 1 false detection that matches the f32 false positive
- LFFD 8 bits has 2 false positives, 1 of them matches f32

False Positives



- 32f
- 16 bit
- 8 bit

- 8 bit

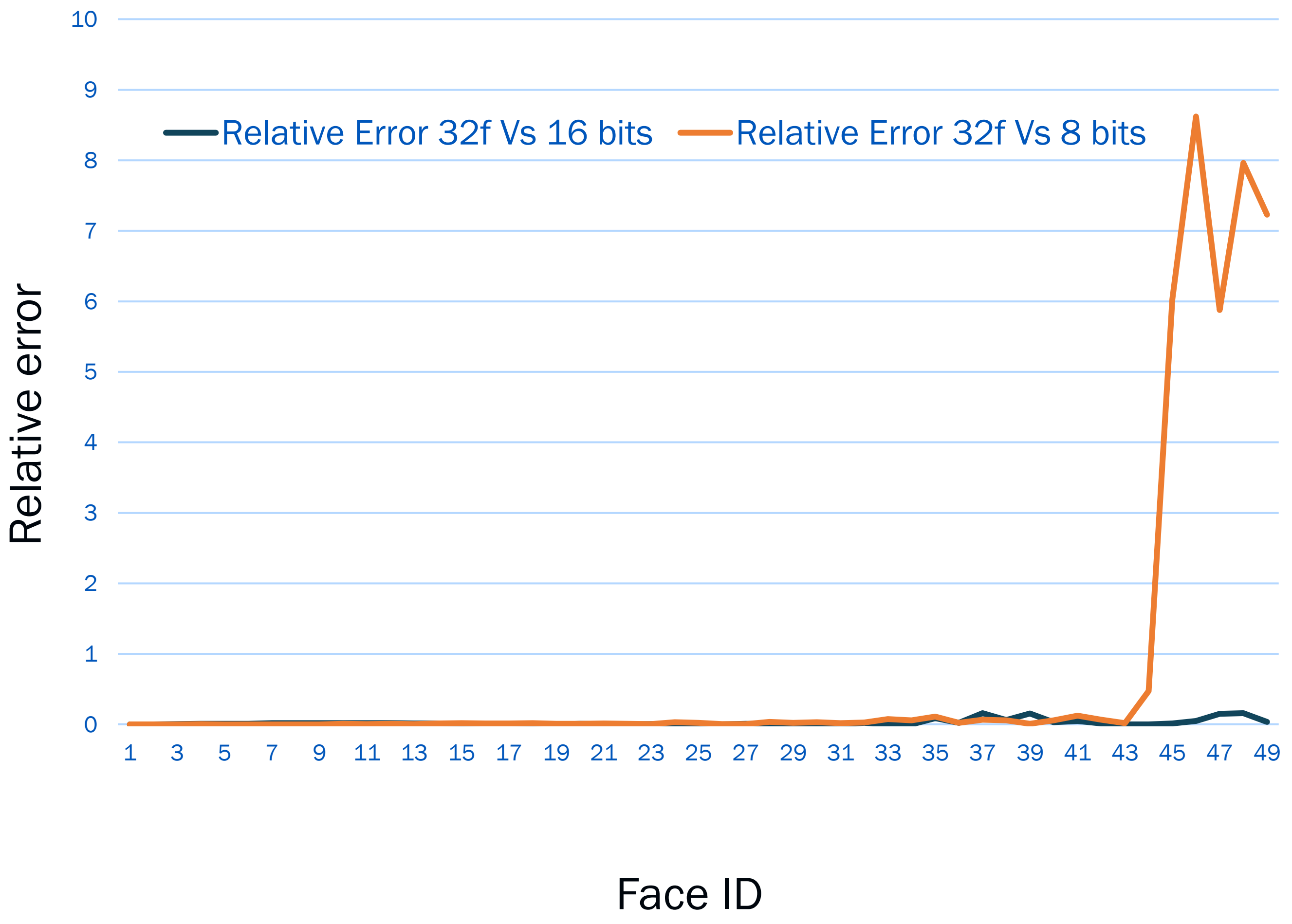
# Face Detection

Face detection boxes match closely between 32f, 16 and 8 bits for LFFD

Relative error graph

- The graph shows the relative error when comparing the confidence of the detection: 32fp, 16 bit and 8 bit.

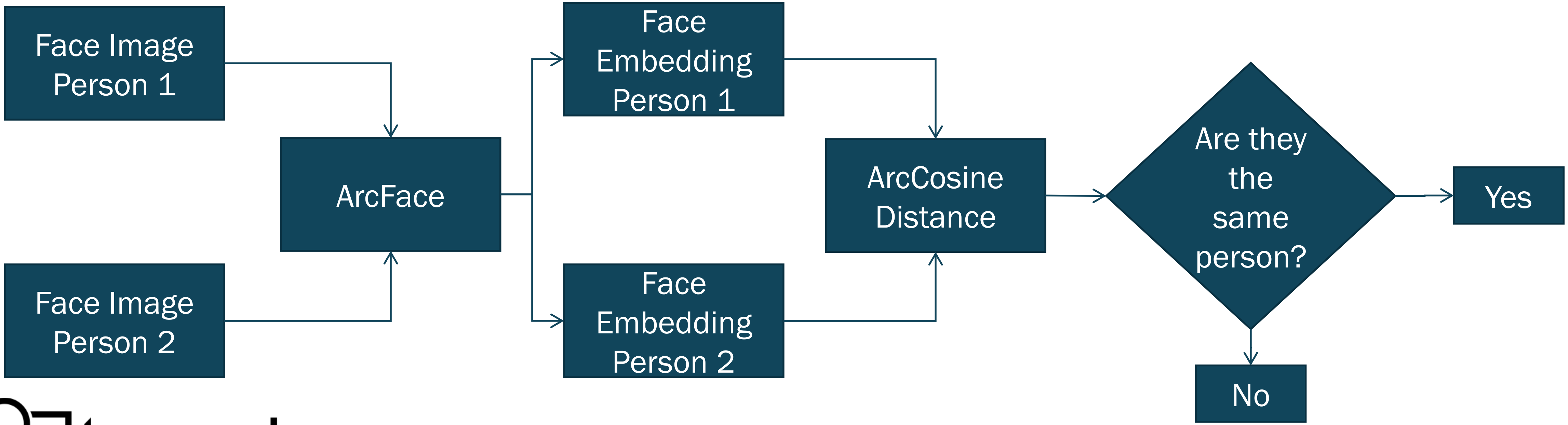
- Face ID's have been sorted according to their distance to the camera.
  e.g. Face ID = 1 is the closest



# Face Verification: ArcFace

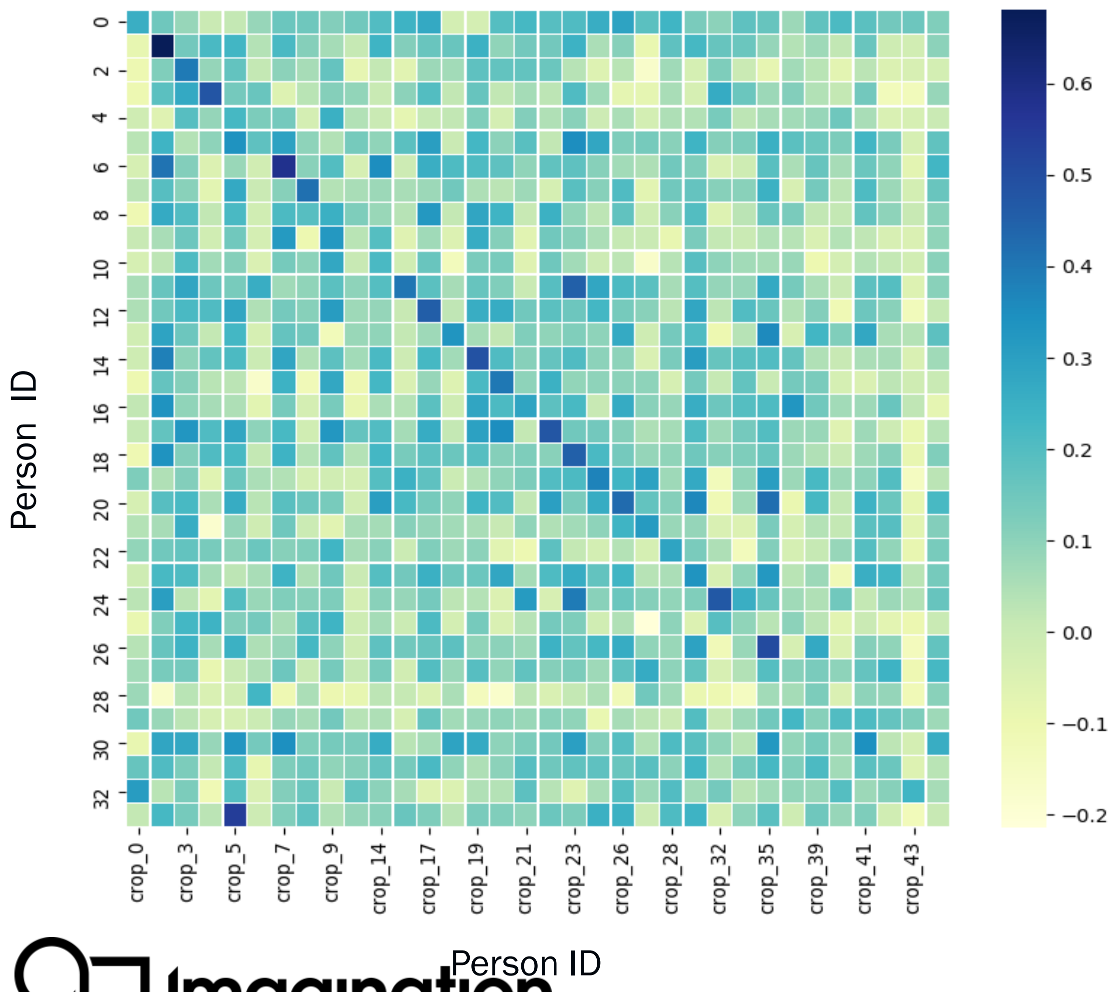
ArcFace is not much about network structure. It is about the loss function, which is used to train a base feature extractor network to produce such features which are a good representation of a face: "embedding“.

There are several different loss functions, such as margin-loss, intra-loss, inter-loss, triplet-loss (introduced in "networks" like CosFace, SphereFace, etc.). The ArcFace paper introduces the Additive Angular Margin Loss.

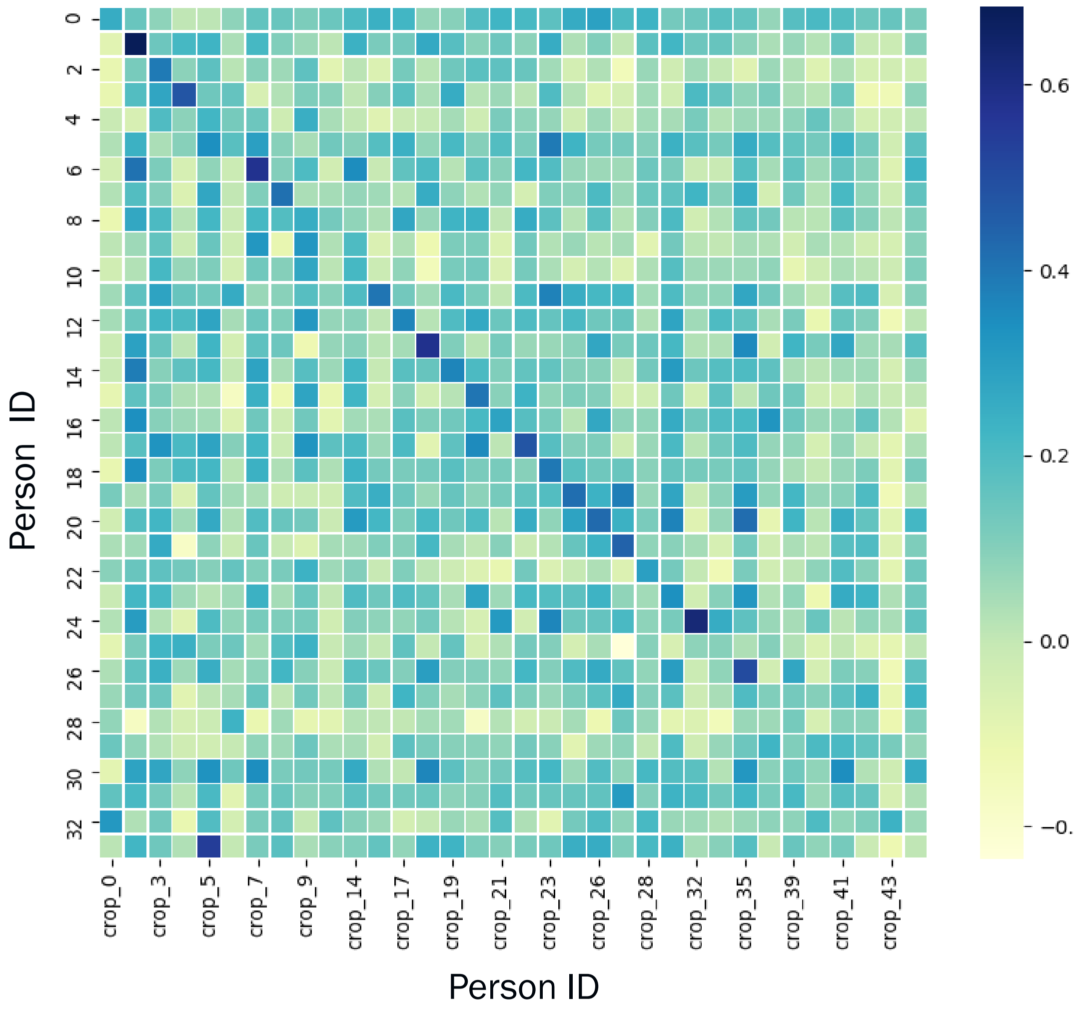


# Face Verification: ArcFace

ArcFace 32f cross validation heatmap, using LFFD for face detection on 8 bits.

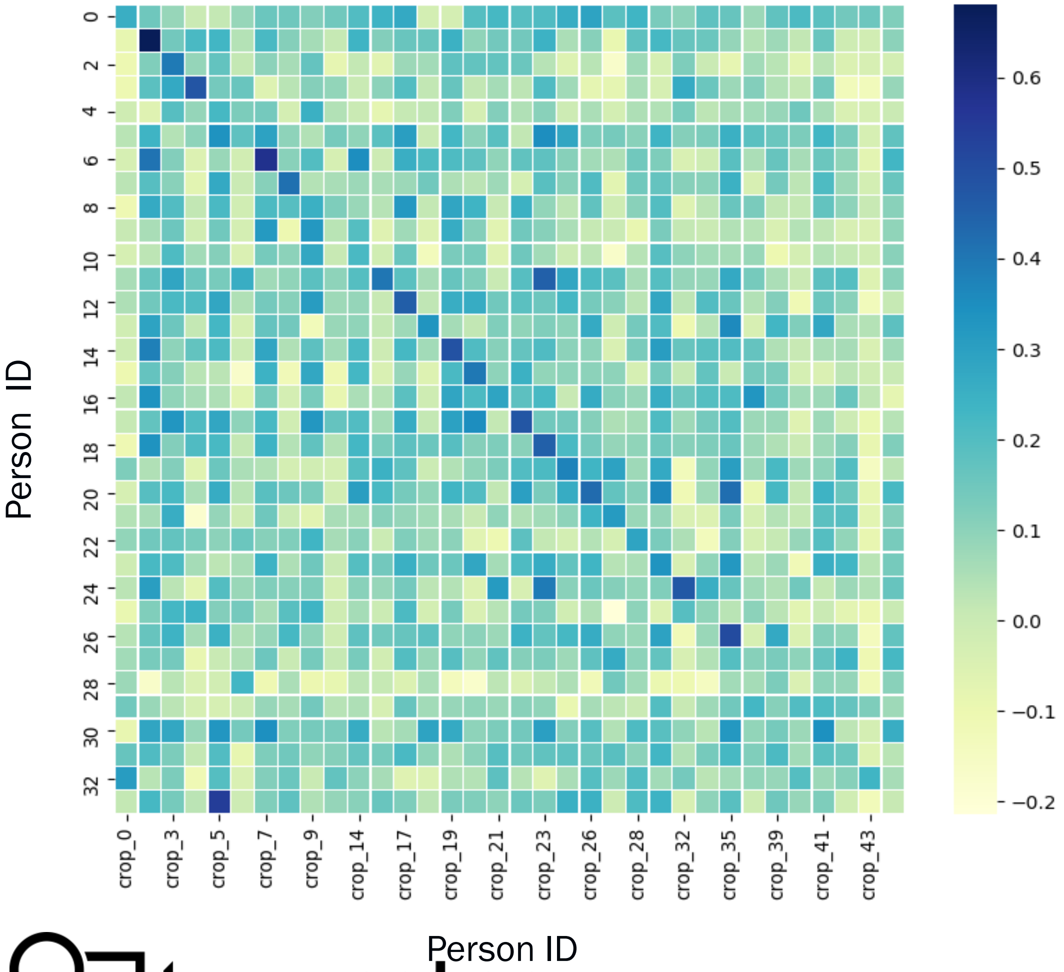



ArcFace 16 bits cross validation heatmap, using LFFD for face detection on 8 bits.

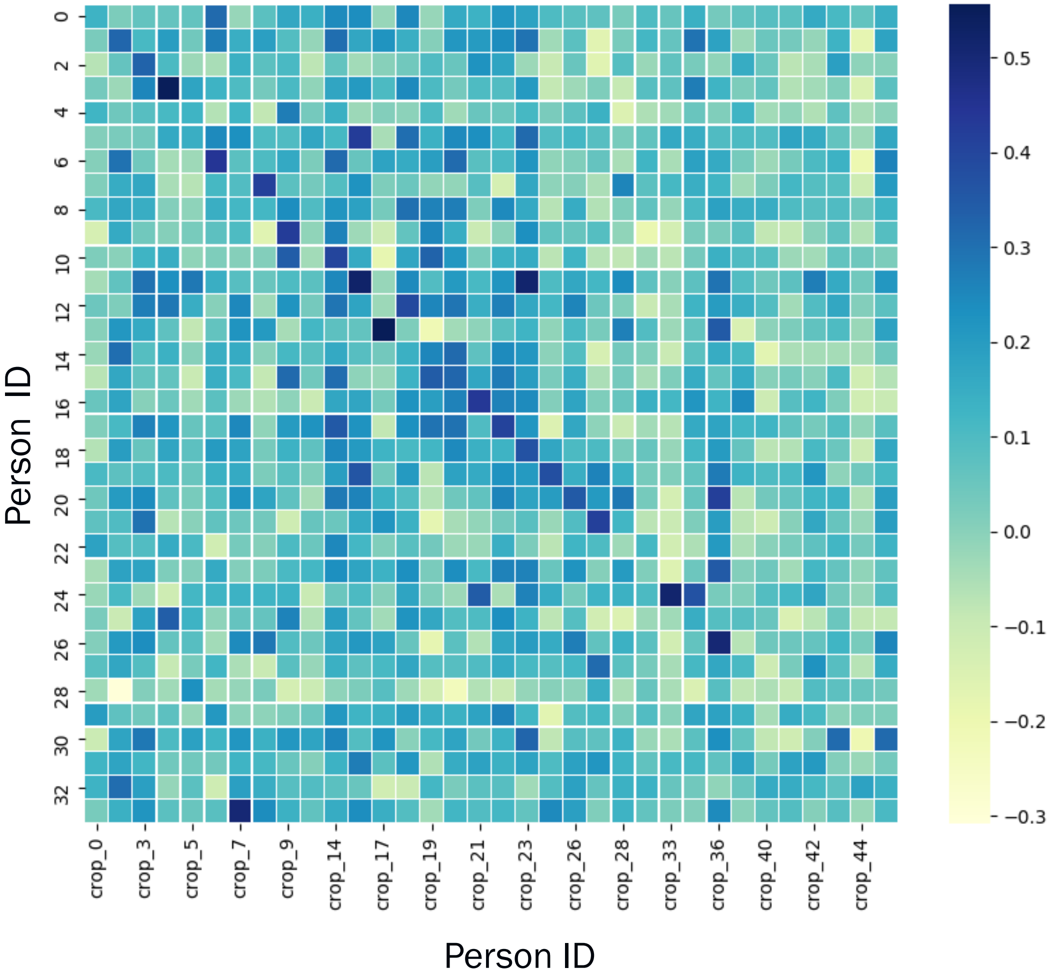


# Face Verification: ArcFace

ArcFace 32f cross validation heatmap, using LFFD for face detection on 8 bits.



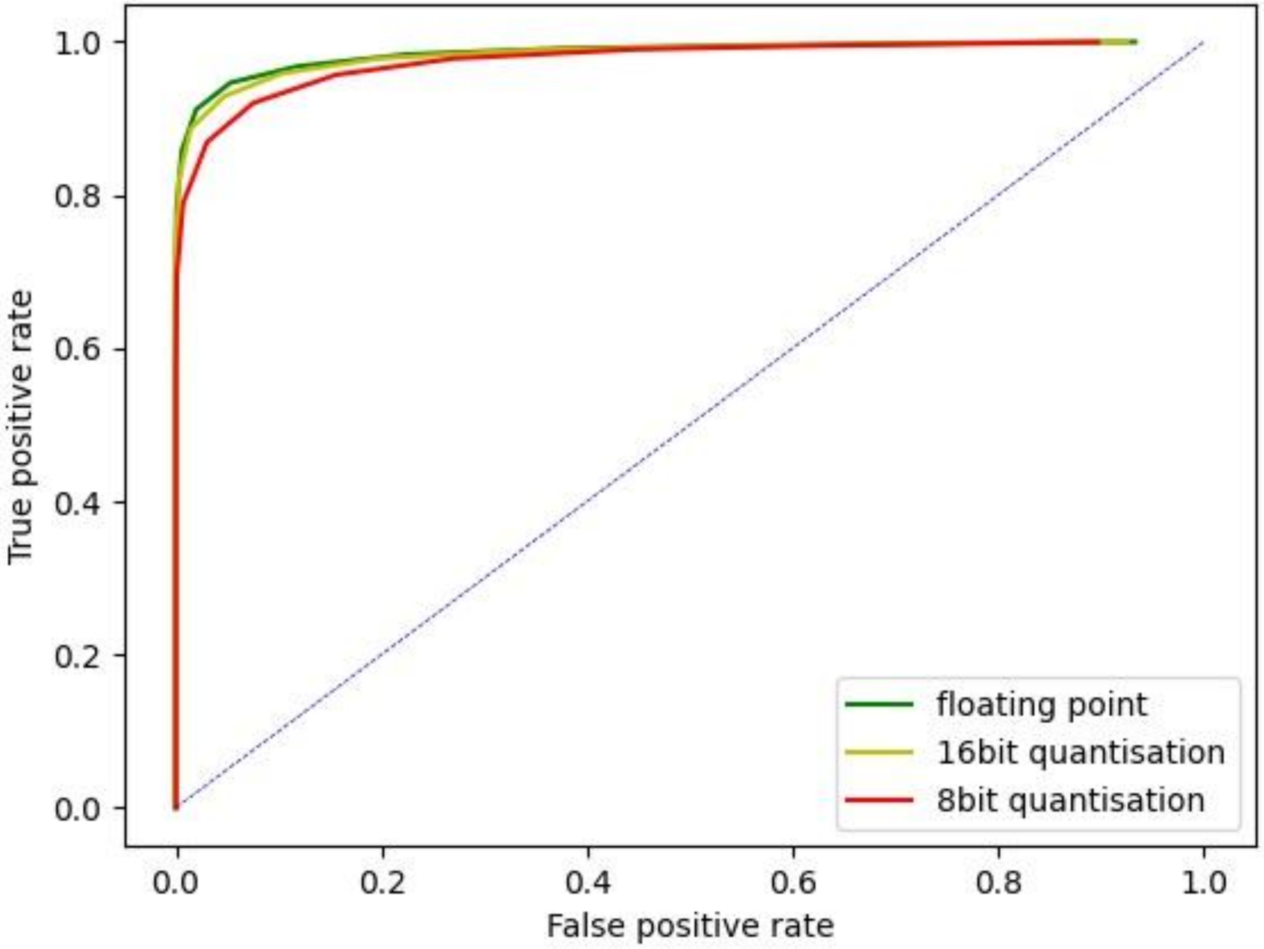

ArcFace 8 bits cross validation heatmap, using LFFD for face detection on 8 bits.



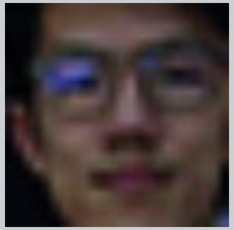
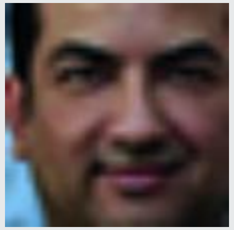
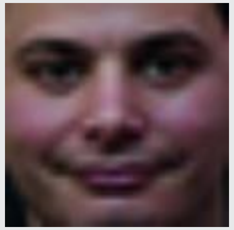
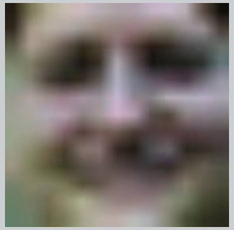
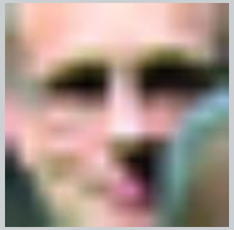
# Face Verification: ArcFace

ROC curve for ArcFace performing on lfw dataset

- Fix point 16 bit quantisation
- Fix point 8 bit quantisation



# Samples correctly classified by different formats

| 32fp | 16bit fix point | 8bit fix point | Examples |
|---|---|---|---|
| ✓ | ✓ | ✓ |      |
| ✓ | ✓ | ✗ |     |
| ✗ | ✗ | ✗ |    |

## Challenges

- Distance to the camera
- Partially occluded face
- Shadows on the face
- Different facial expressions with respect to the ground truth

# Performance

PowerVR 2NX

| | Bandwidth per inf (MB) | | Inferences per seconds | |
|---|---|---|---|---|
| | 16 bits | 8 bits | 16 bits | 8 bits |
| LFFD (640x640) | 260 | 112 | 56 | 83 |
| ArcFace (112x112) | 206 | 95 | 70 | 123 |

PowerVR 3NX

| | Bandwidth per inf (MB) | | Inference per seconds | |
|---|---|---|---|---|
| | 16 bits | 8 bits | 16 bits | 8 bits |
| LFFD (640x640) | 218 | 101 | 65 | 158 |
| ArcFace (112x112) | 170 | 73 | 89 | 242 |

# Take away

- Based of our results 8 bit quantisation is not a recommended option for reliable face verification for low resolution images.
  Options:
  - different data format e.g. 16 bit fixed point
  - Per channel quantisation

- LFFD was designed specifically for embedded systems.
- In the case of ArcFace: Channel-wise operations can cause extra challenge for hardware designs due to per-channel data or per-channel logic requirement.

- Tailoring a NN for embedded systems is a good alternative to take the best of an architecture.

- The face verification embedding needs to be calculated for each individual face. The compute demand can be big compare to the face detection workload. This depends of the number of faces detected and the type of verification.
  Options:
  - A scalable High Performance Compute architecture
  - A face verification design that reuse the feature maps of the detections boxes.

# References


- DeepLab V3: Rethinking Atrous Convolution for Semantic Image Segmentation
  https://arxiv.org/pdf/1706.05587.pdf

- LFFD: A light and fast face detector for edges devices
  https://arxiv.org/pdf/1904.10633.pdf

- ArcFace: Additive Angular Margin Loss for Deep Face Recognition
  https://arxiv.org/pdf/1801.07698.pdf

# Thank you

# Questions

Imagination