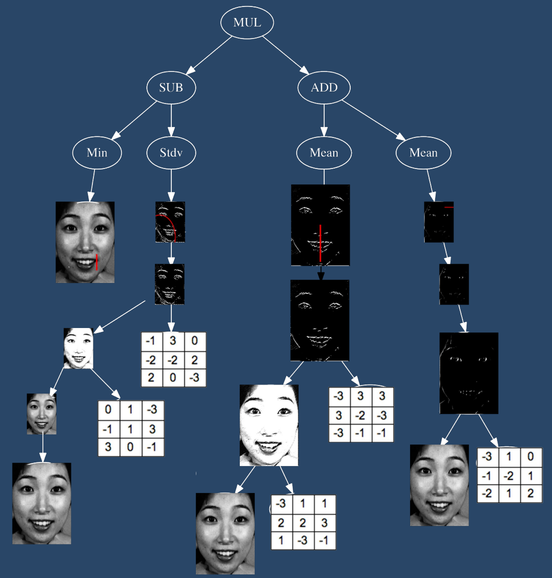
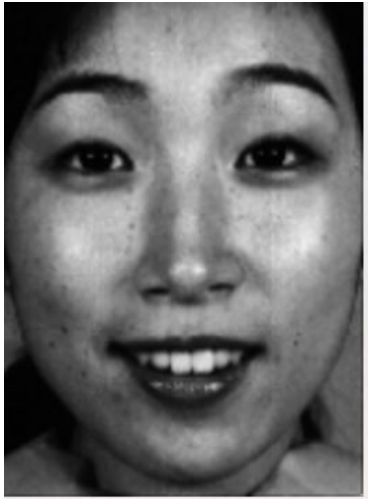


ConvGP

Genetic Programming for Evolutionary Deep Learning for Image Classification

Ben Evans | Mengjie Zhang, Bing Xue, Harith Al-Sahaf

Goal?

A novel method for Binary Image Classification
which overcomes some major current limitations

Specific Objectives

1

Develop a novel deep GP Structure

That can correctly classify images by automatically detecting interesting regions from the images, performing feature extraction on these regions, and then automatically constructing higher level features from these extracted features.

2

Visualisation

Visualise and interpret the features automatically extracted, constructed and learned by the evolutionary learning process.

3

Feature Construction

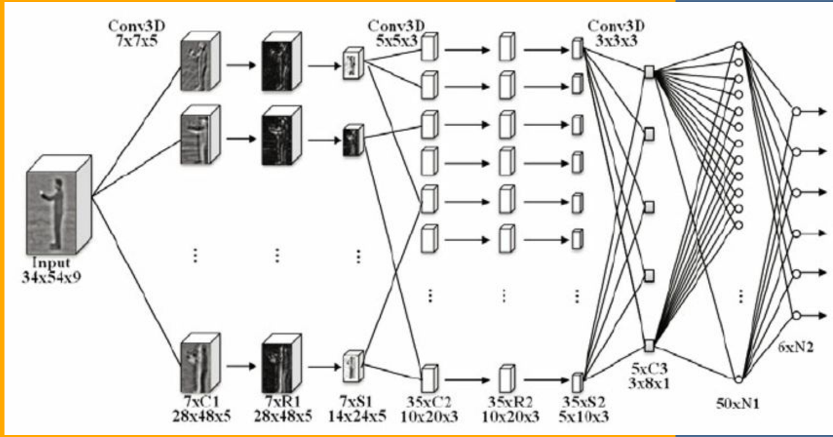
Investigate whether the automatically extracted/constructed features can be useful for common classification algorithms, such as Nearest Neighbour or Decision Trees

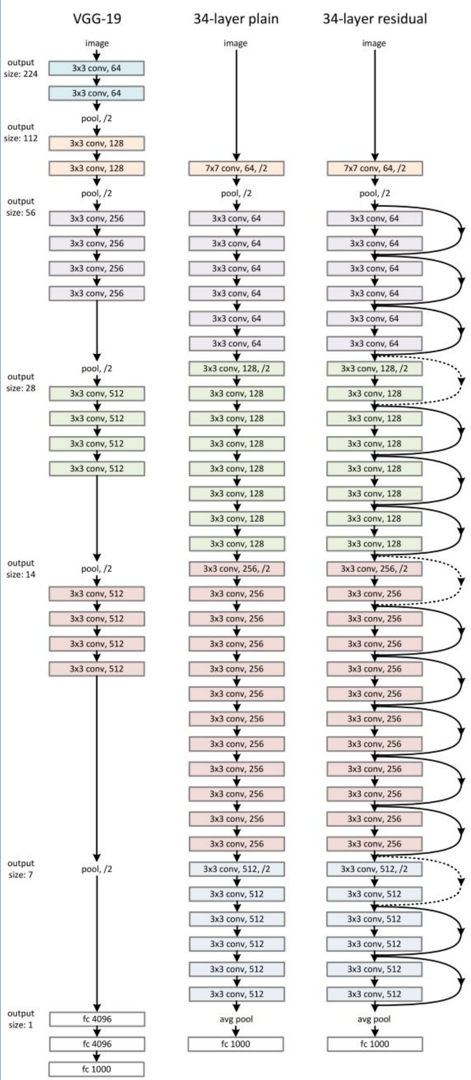
Motivations

Convolutional Neural Networks

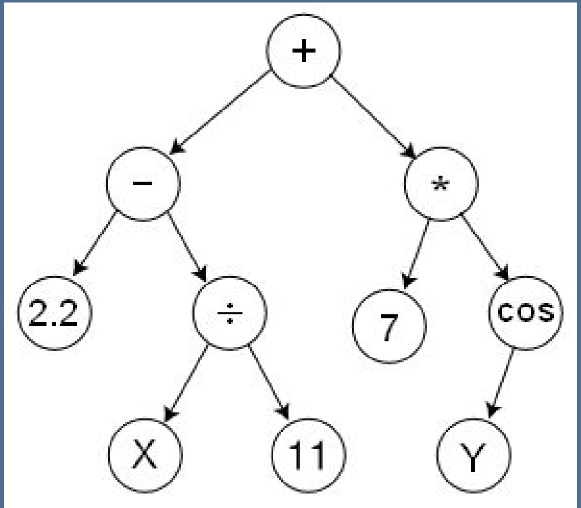
Achieve great accuracy, however suffer some limitations

- Manually crafted architectures
- Poor interpretability
- Need for large amounts of training data





Genetic Programming



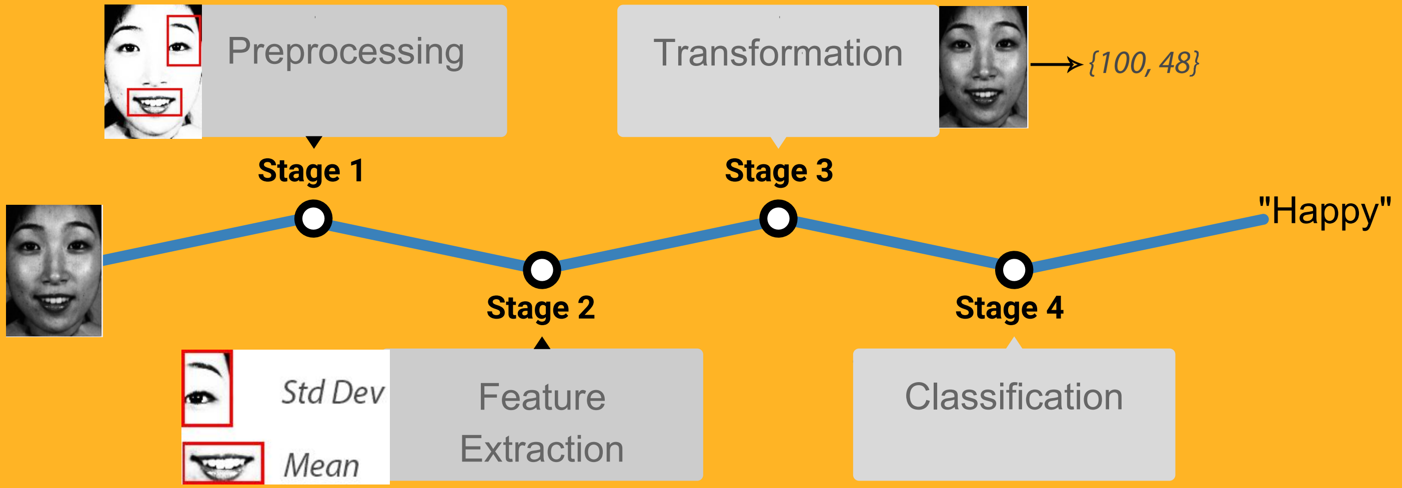
Genetic Programming

Accuracy not currently as high as conv net methods. However has some other benefits such as

- Greater Interpretability
- Automatic programming
- Automatically determine appropriate height/width
- Do not require domain knowledge (although can be utilised)

However they are not specifically defined for image classification or widely used for feature maps

Image Classification Design Decisions and Constraints



Challenges

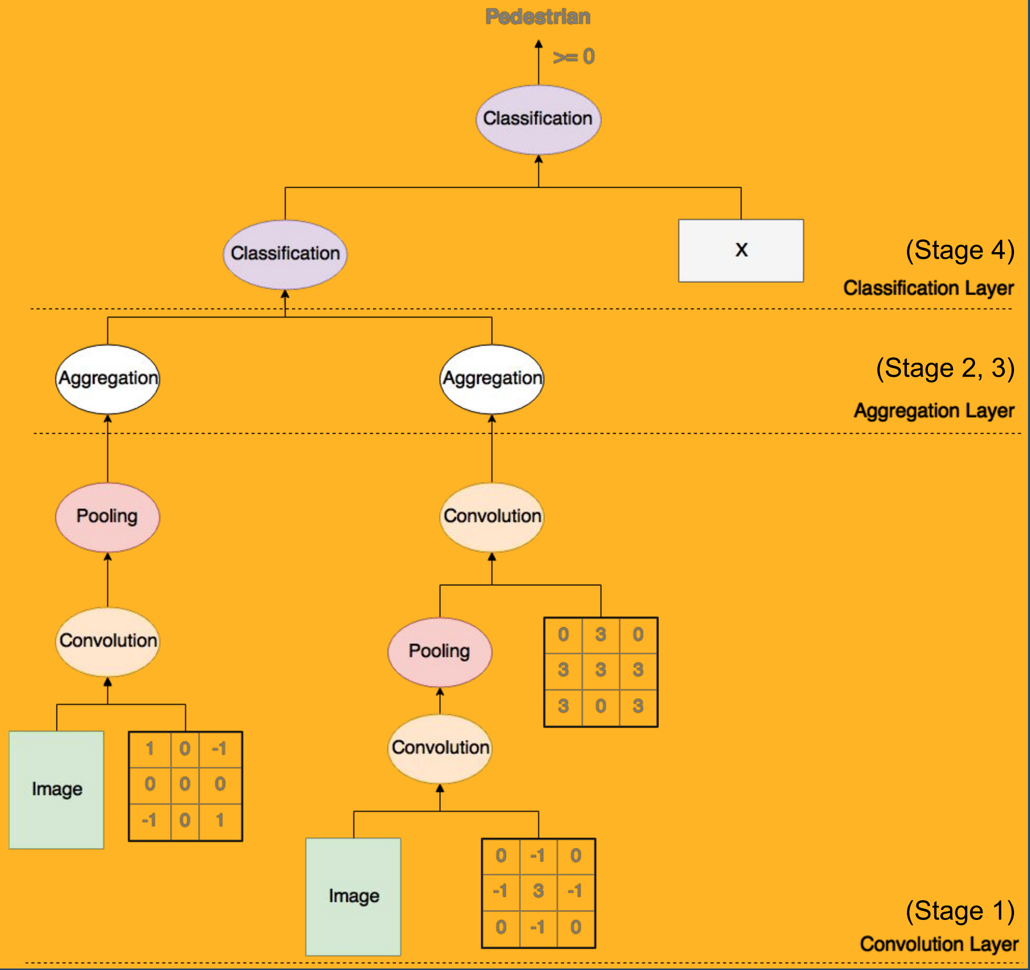
How to remove the need for human intervention?

How can the various stages be combined into a single algorithm?

How to incorporate domain knowledge, i.e. neighbouring pixels are related?

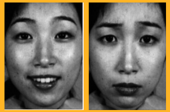
How to incorporate the useful components from convnets and genetic programming?

New Method
Incorporates Key ideas from both

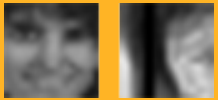


Experiment Design

Datasets


JAFFE

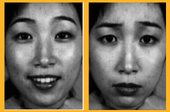

Cars


Faces


Pedestrians

Experiment Design

Datasets



JAFFE

Cars

Faces

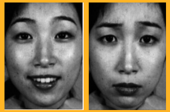
Pedestrians

Benchmark Methods

- ConvNet
- Existing GP approach (two-tier GP)
- Decision Trees
- Naive Bayes
- Nearest Neighbour
- Adaboost
- Support Vector Machine

Experiment Design

Datasets



JAFFE

Cars

Faces

Pedestrians

Benchmark Methods

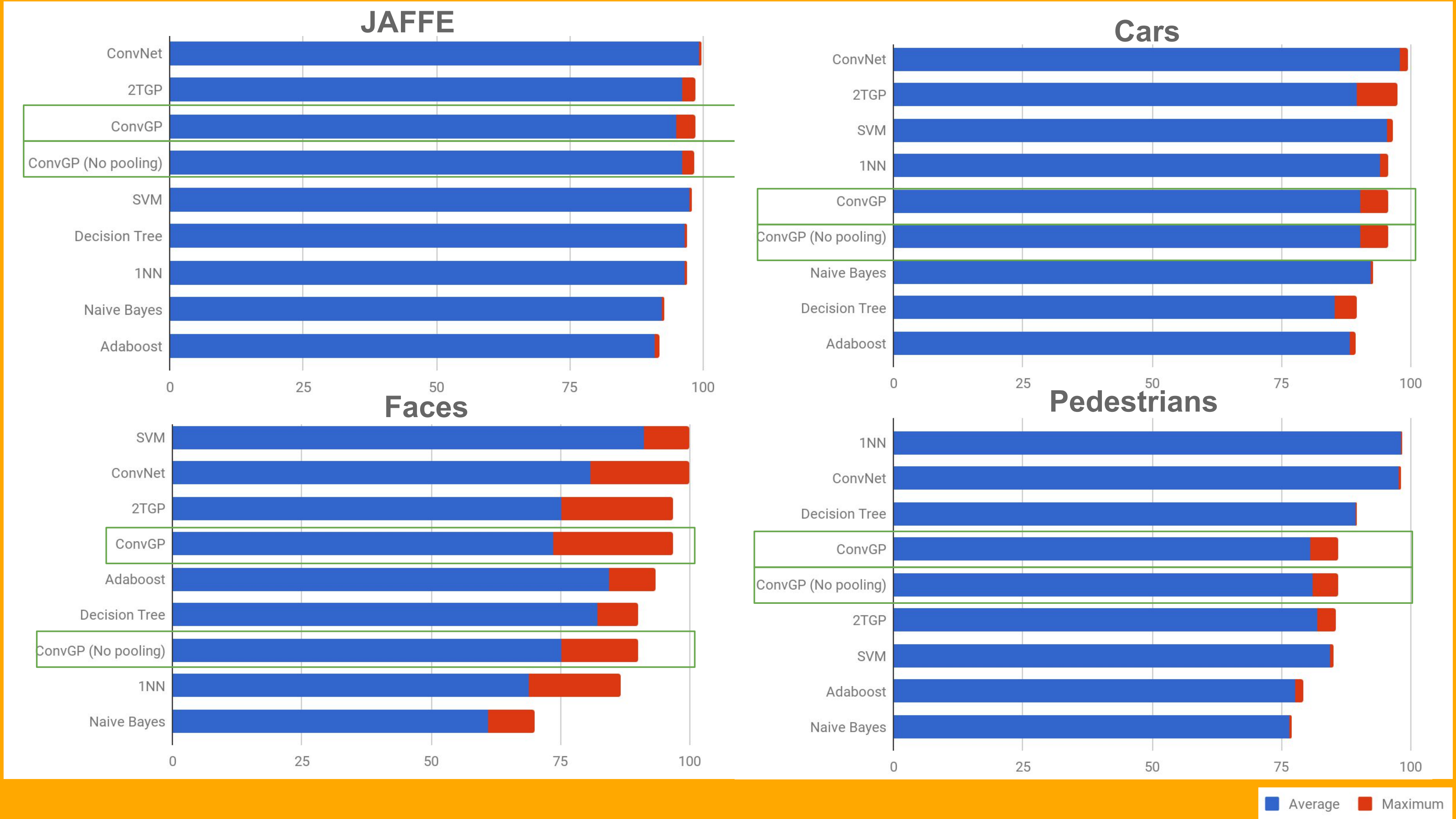
- ConvNet
- Existing GP approach (two-tier GP)
- Decision Trees
- Naive Bayes
- Nearest Neighbour
- Adaboost
- Support Vector Machine

Parameter Settings

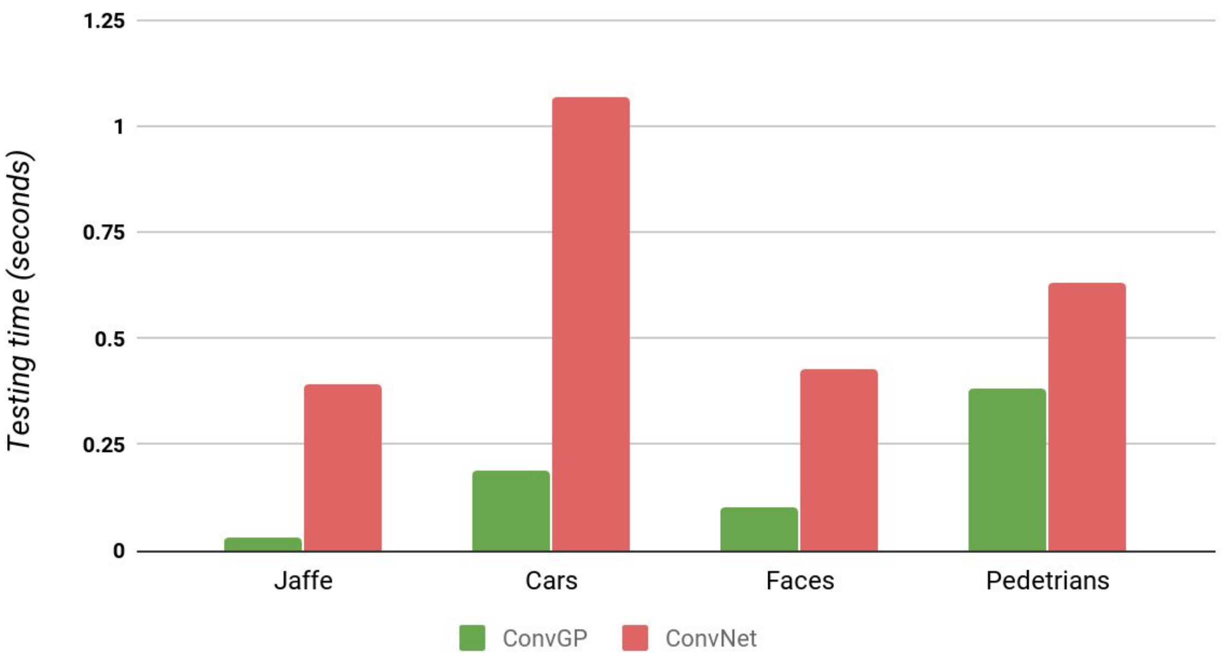
Population	1024
Generations	50
Max Depth	10
Tournament	7
Crossover	0.8
Mutation	0.2
Elitism	0.01

Results

Results: Average Testing Accuracy



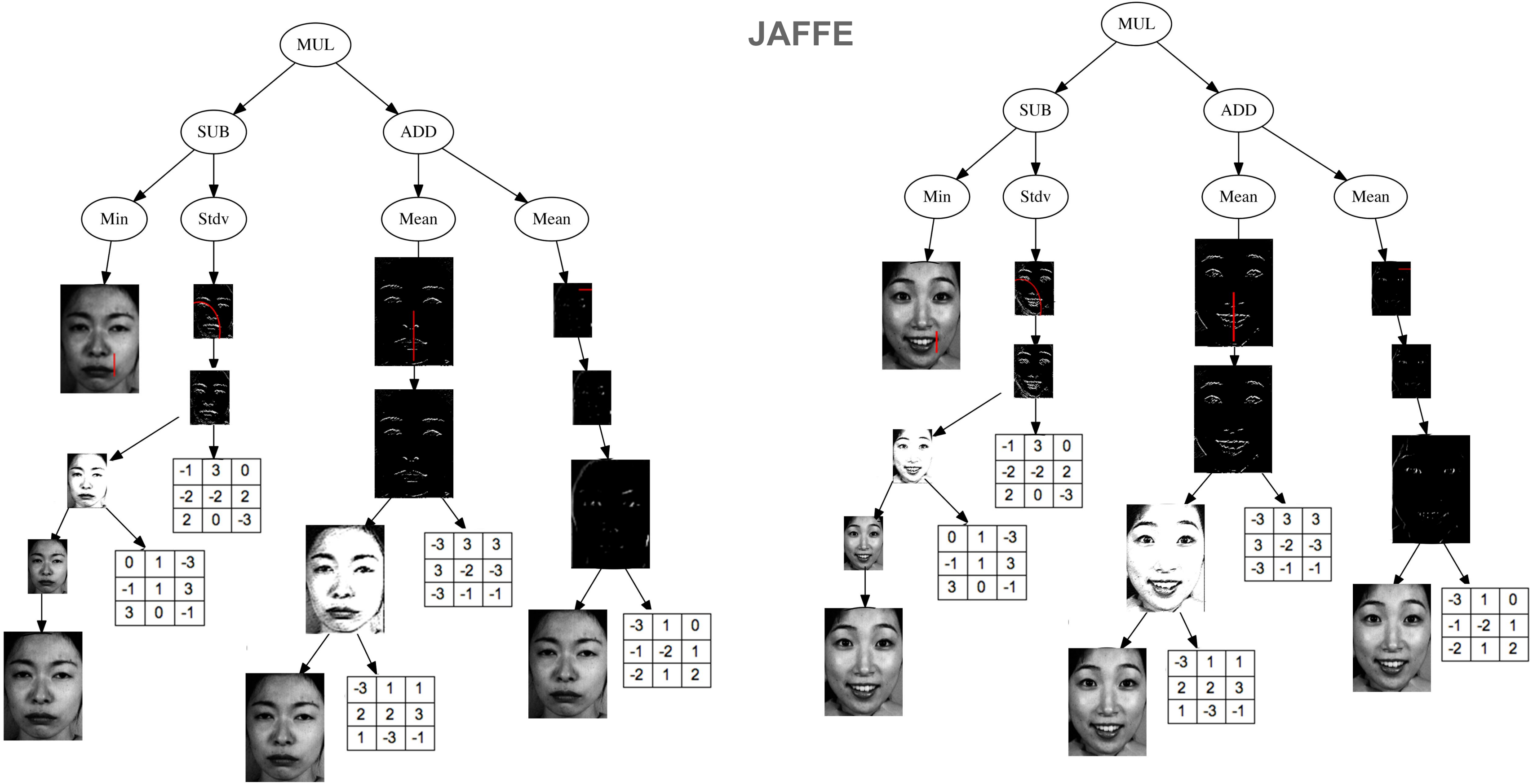
Results on Efficiency: Average Testing Time vs Conv Nets



*Note: the time taken is the total time for all images in the testing set

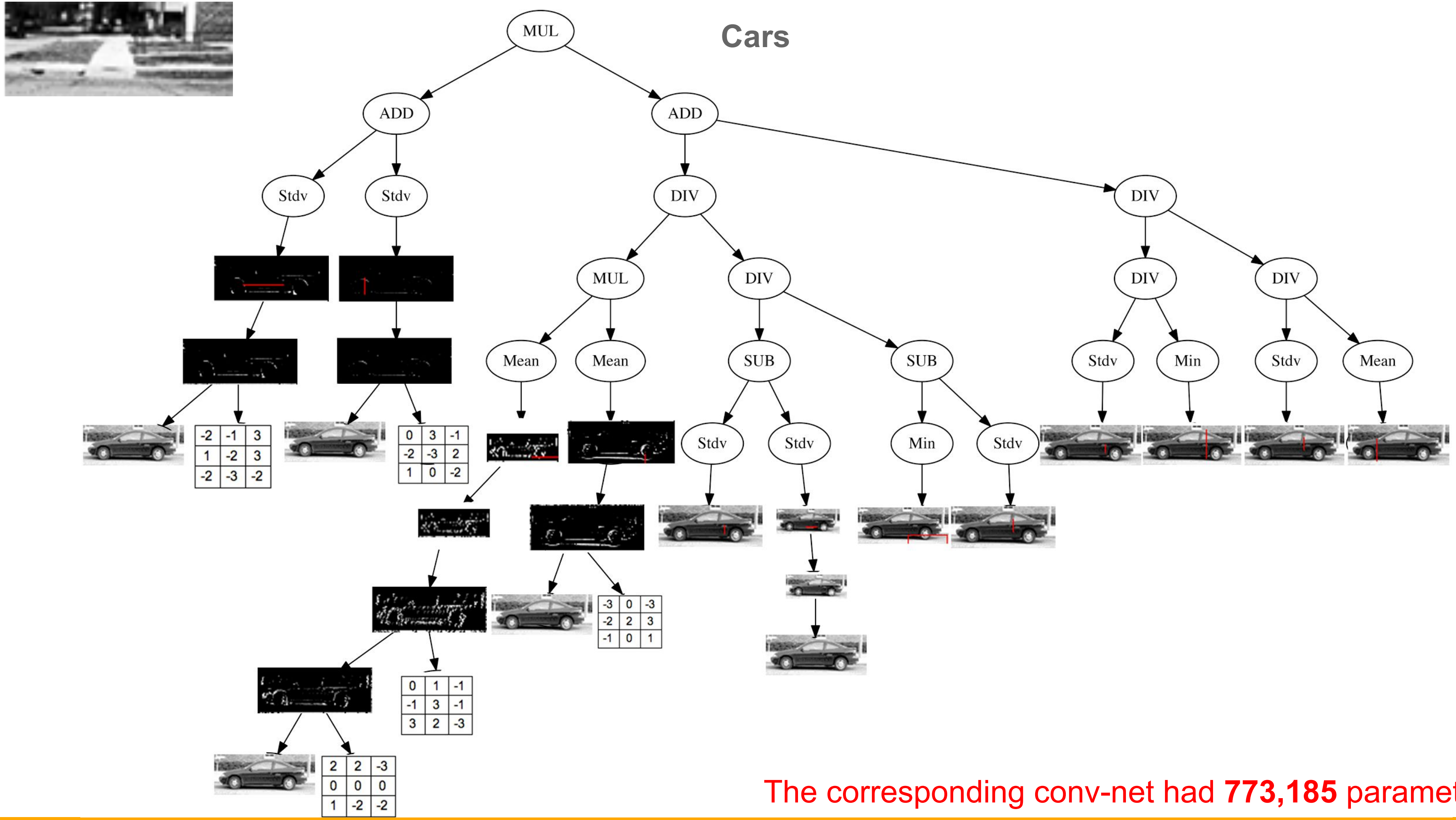
Visualisation

JAFFE



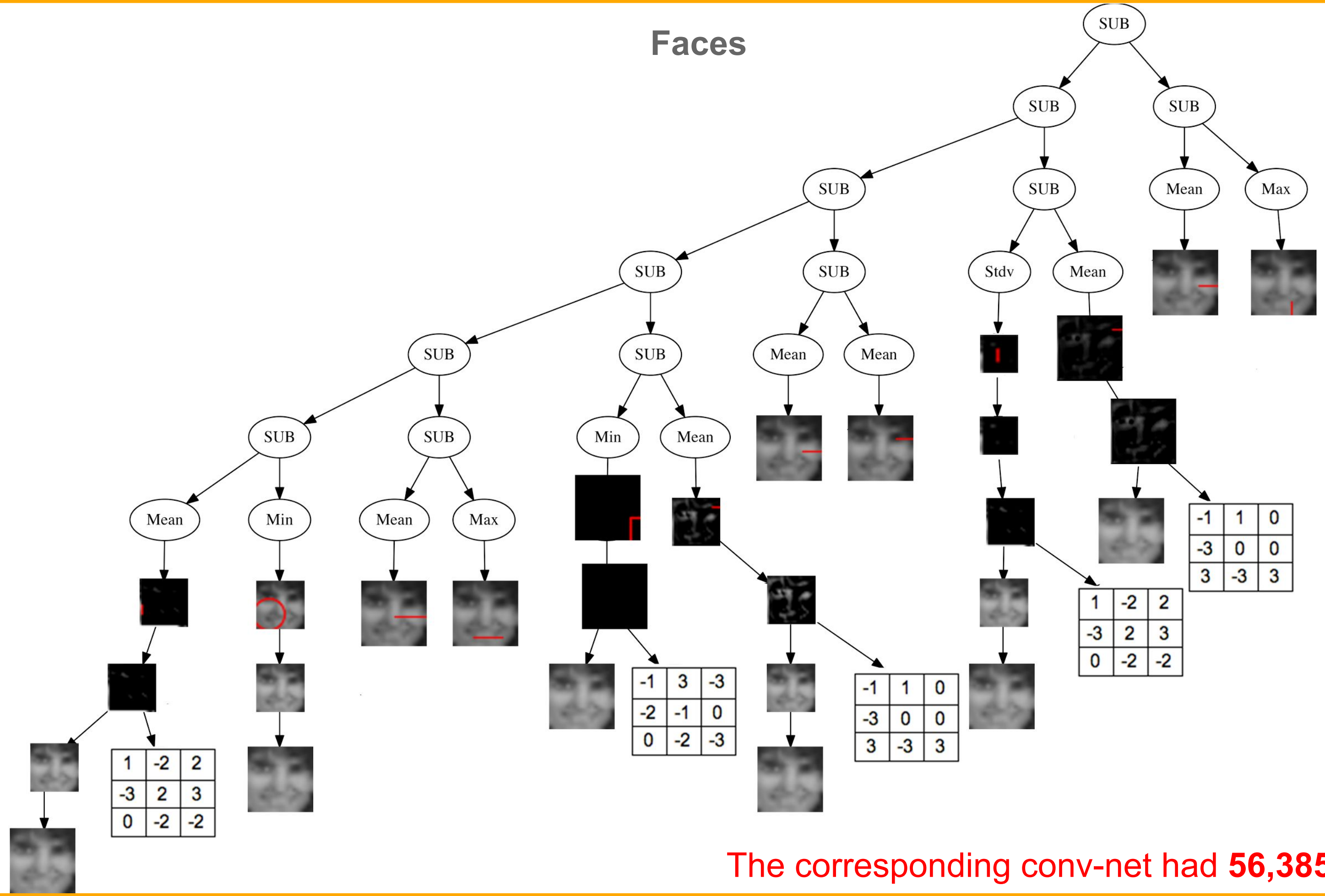
The corresponding conv-net had **5,479,489** parameters

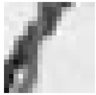
Cars



The corresponding conv-net had **773,185** parameters

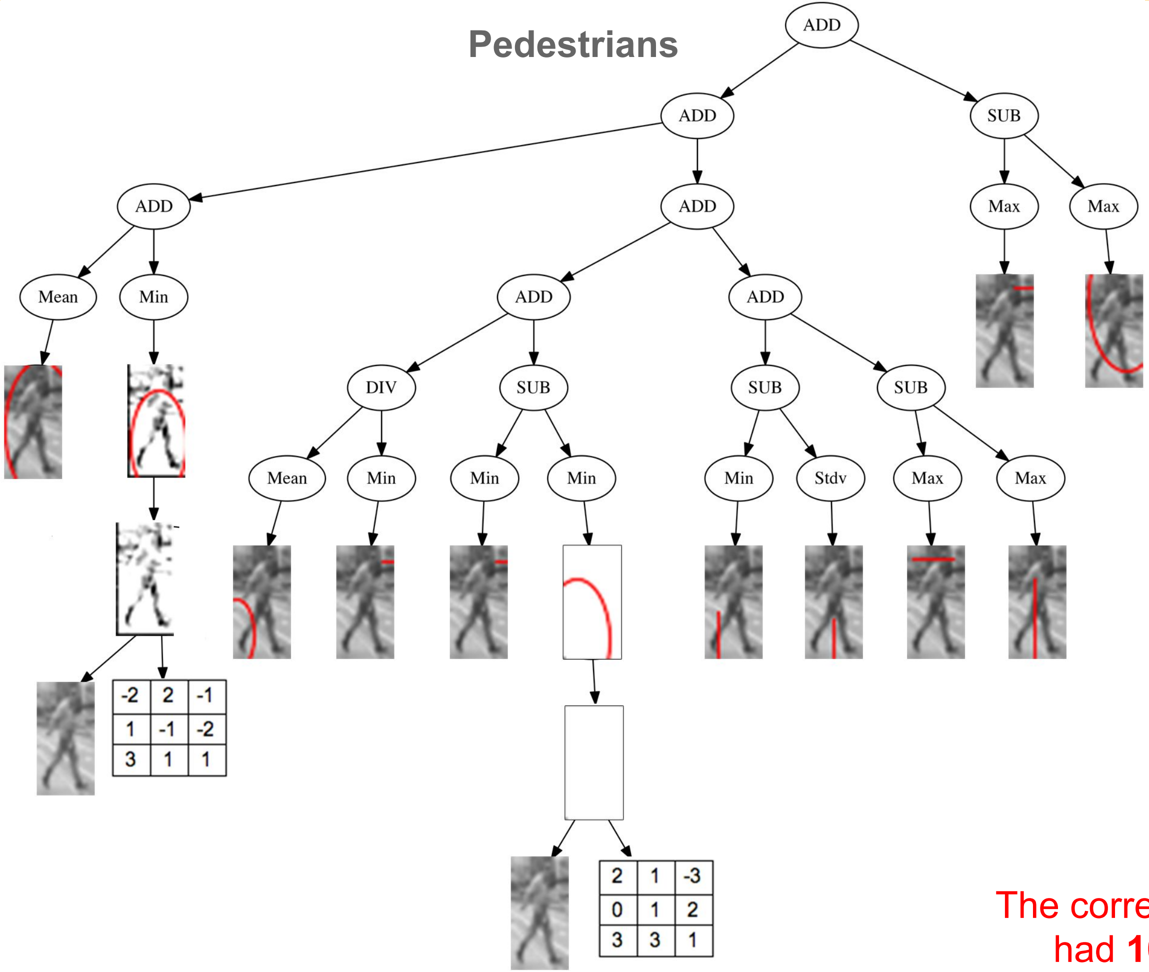
Faces





The corresponding conv-net had **56,385** parameters

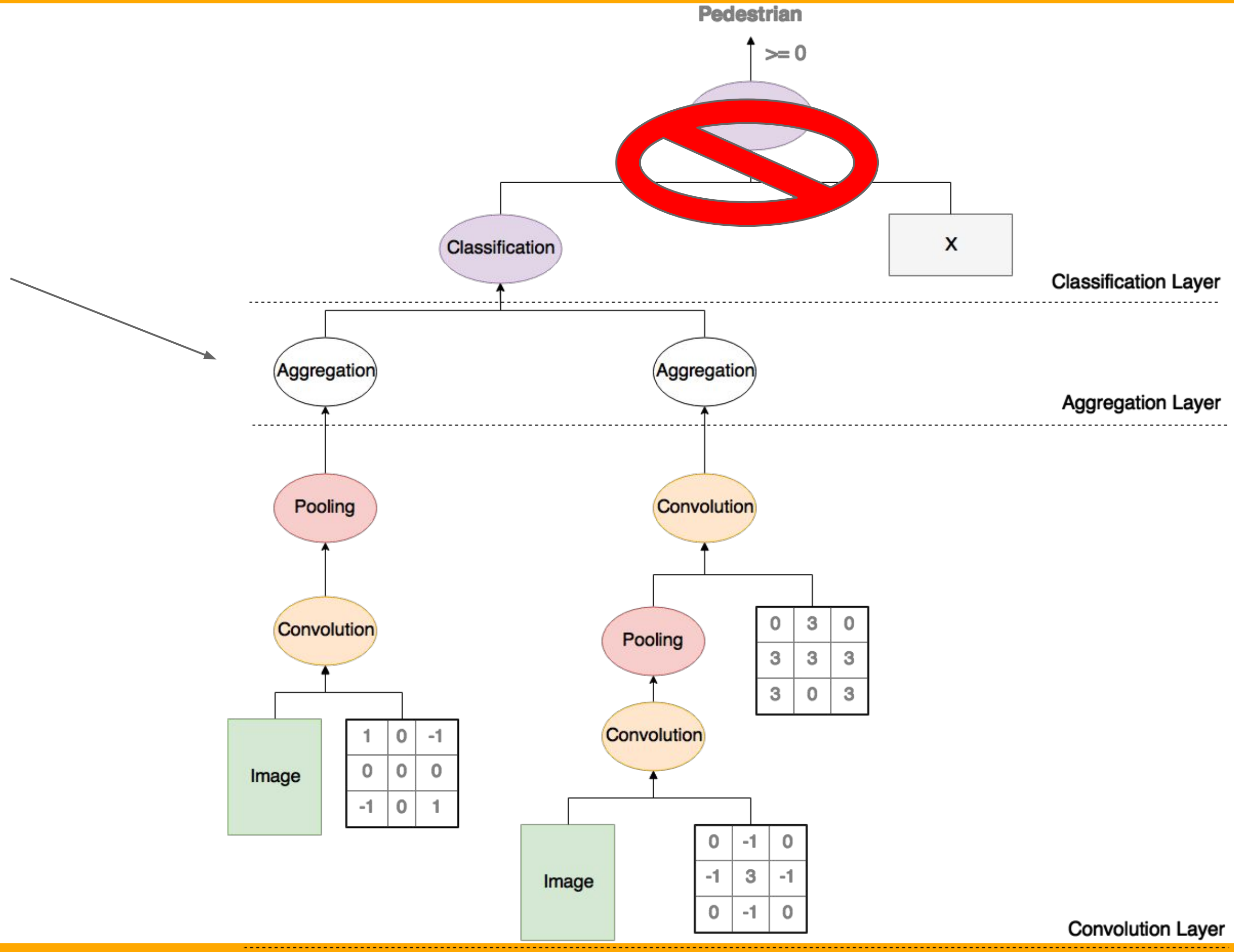
Pedestrians



The corresponding conv-net
had **105,537** parameters

Feature Construction

Set of constructed features from output of aggregation tier



Dimensionality Reduction

	Original		Average # of Constructed
Jaffe	23,400	→	**8**
Cars	4,000	→	**13**
Faces	361	→	**14**
Pedestrians	648	→	**14**

Highlights of feature construction results

Jaffe			
	Training Speedup	Testing Speedup	Testing Accuracy
Naive Bayes	32x	22x	**Improvement**
Nearest Neighbour	7x	8x	**Improvement**

Cars			
	Training Speedup	Testing Speedup	Testing Accuracy
Naive Bayes	263x	284x	**Improvement**
Decision Tree	220x	7x	**Equivalent**

Faces			
	Training Speedup	Testing Speedup	Testing Accuracy
Naive Bayes	23x	33x	**Equivalent**
Nearest Neighbour	14x	18x	**Equivalent**

Pedestrians			
	Training Speedup	Testing Speedup	Testing Accuracy
Naive Bayes	42x	43x	**Improvement**
Adaboost	30x	3x	**Equivalent**

Note: Testing accuracy column based on an unpaired welch t-test on resulting testing accuracies. **Improvement** means statistical improvement ($p < 0.05$) with the constructed features, **Equivalent** means no statistically significant difference ($p > 0.05$).

Conclusions

1. Classification: The proposed method was able to outperform many general classification methods in terms of maximum testing accuracy.

In terms of classification accuracy, although the method was unable to outperform ConvNets, the proposed solution offered **faster testing**, **greater interpretability** and the architecture was able to be evolved **automatically rather than manually** crafted.

Objectives

1. Develop a novel structure
2. Visualisation
3. Feature Construction

Conclusions

2. Visualisation: The method offered greater visualisation ability than the state-of-the-art convolutional neural networks, which are very difficult to visualise due to the large number of parameters.

Objectives

1 ~~Develop a novel structure~~

2 Visualisation

3 Feature Construction

Conclusions

3. Feature Construction: The constructed features offered

- Drastic reduction in dimensionality
- Matched (and in some cases improved) classification accuracy*
- Huge speedup for both training and testing times

* On average across the various datasets and methods trialled

Objectives

1. ~~Develop a novel structure~~
2. ~~Visualisation~~
3. Feature Construction

Future Work

- Fitness function to learn from a small number of instances
- Architectural changes (half from proposed method, half without convolution tier)
- Look to improve training time/efficiency of process

Thanks!

Any questions?