Sharing Visual Features for Multiclass and Multiview Object Detection

presented by: Stefan de Konink

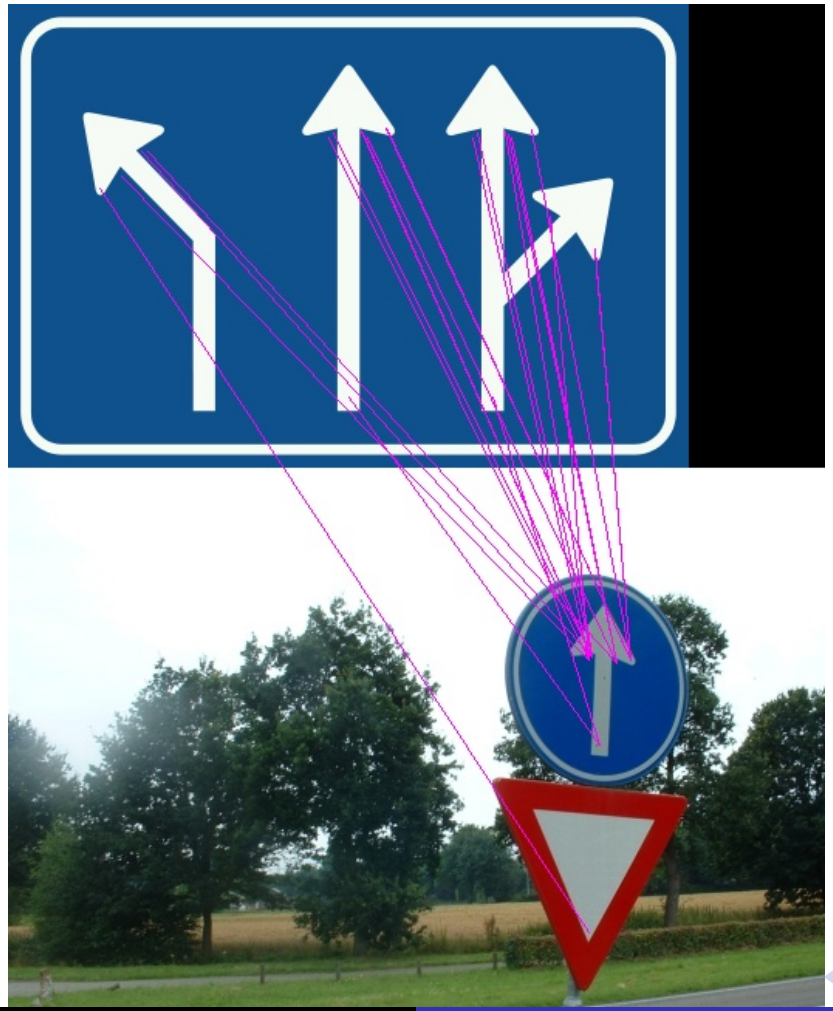
November 27, 2008

Cluttered Scenes

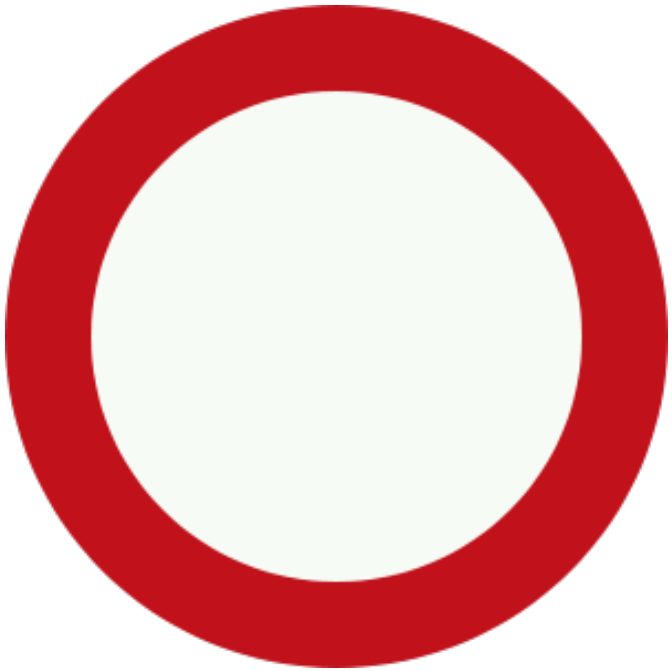
Problems

- Background/Foreground selection
- Performance

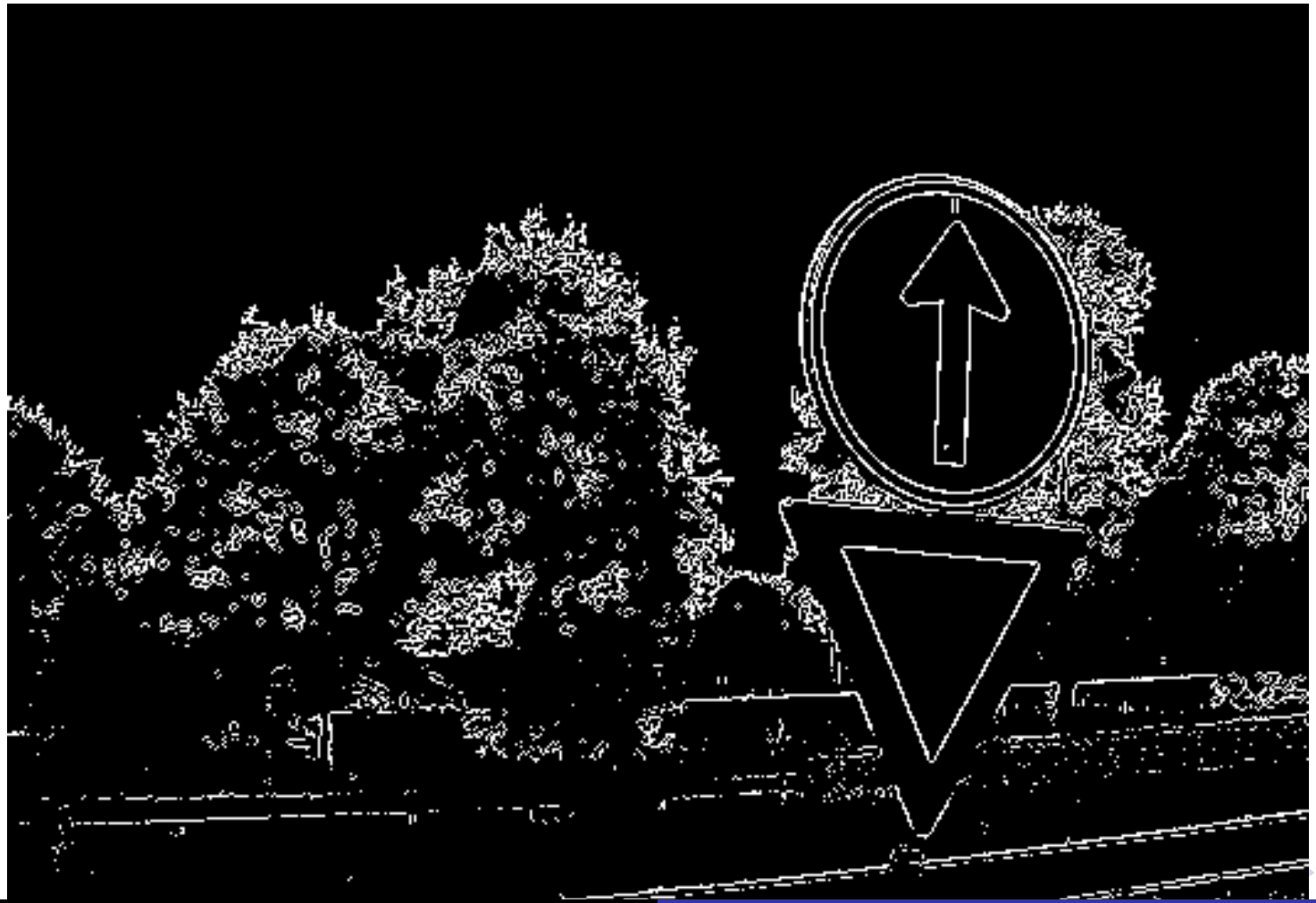
Scale-invariant feature transform



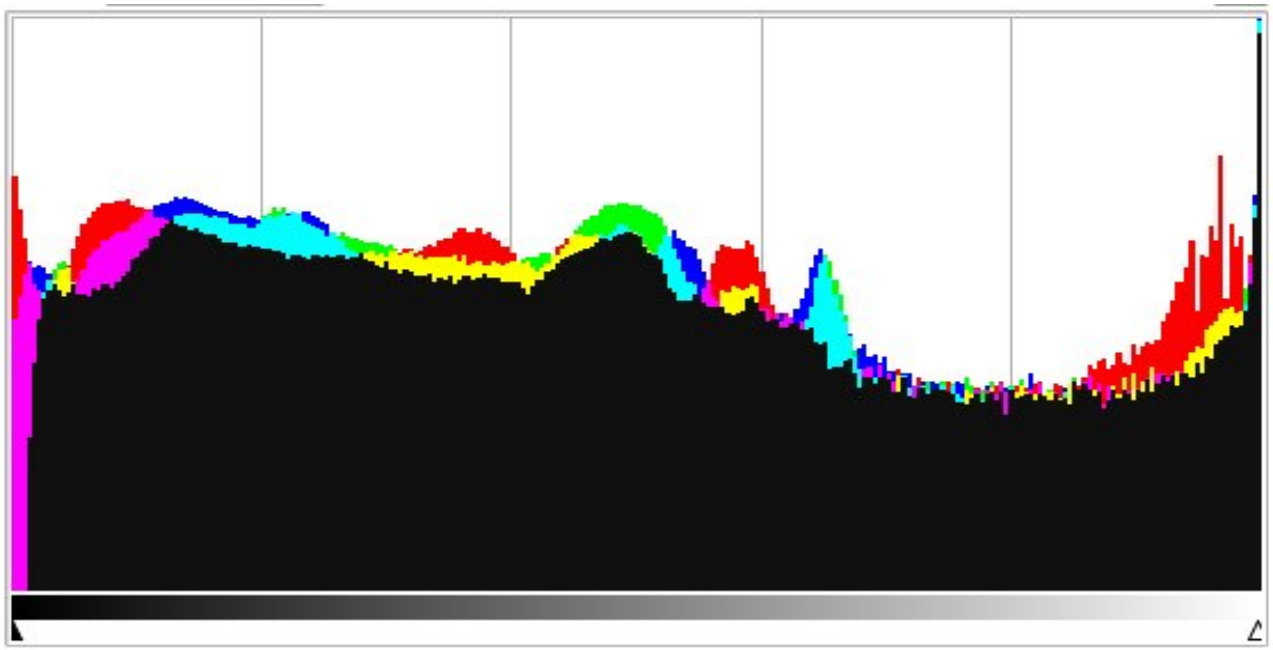
Scale-invariant feature transform



Contour based approaches



Histograms



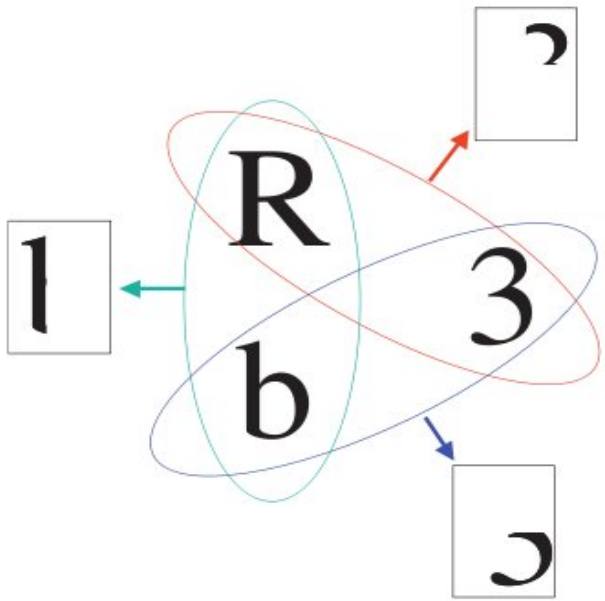
Current work

- Object specific features
- Histogram, contour based approaches

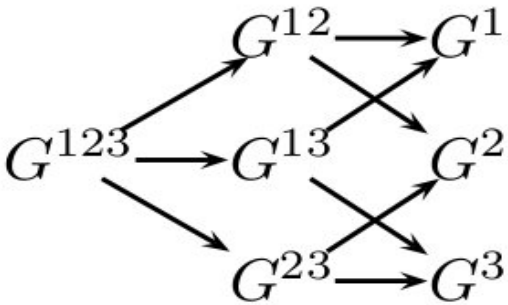
Proposals

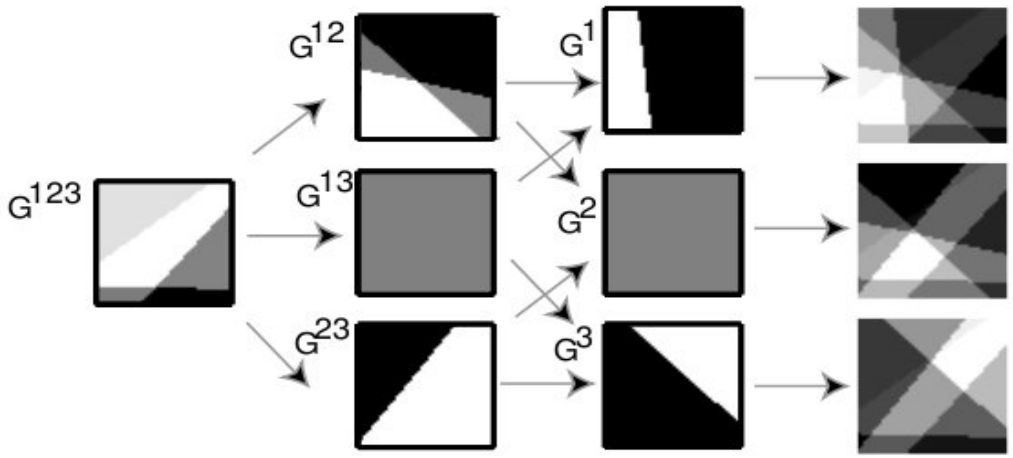
- Share visual features
- Use joint-boosting and weak classifiers

What to share?



How to share?



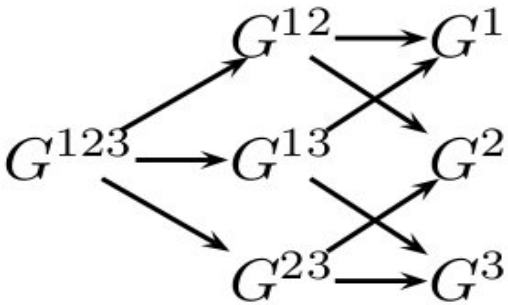


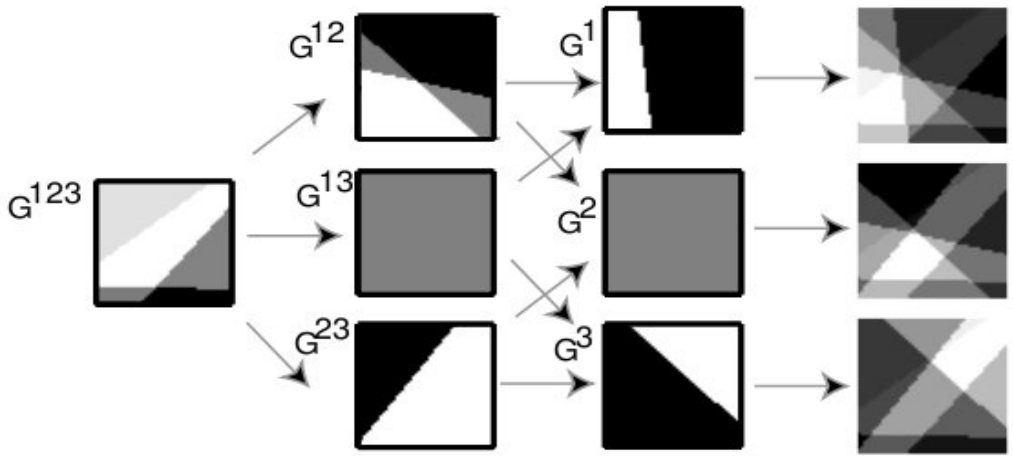
How to share?

$$
\begin{aligned}
H(v,1) &= G^{1,2,3}(v) + G^{1,2}(v) + G^{1,3}(v) + G^{1}(v) \\
H(v,2) &= G^{1,2,3}(v) + G^{1,2}(v) + G^{2,3}(v) + G^{2}(v) \\
H(v,3) &= G^{1,2,3}(v) + G^{1,3}(v) + G^{2,3}(v) + G^{3}(v)
\end{aligned}
$$

Three weak learners (G) shared across classes,
with their classifiers (H).

How to share?





Shared visual features

- Reduces the amount of training data required
- Increases performance

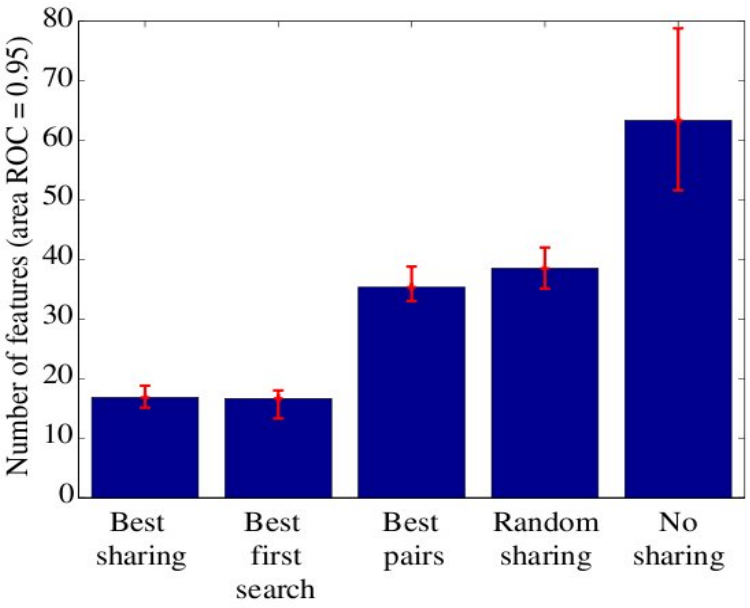
Joint-Boosting

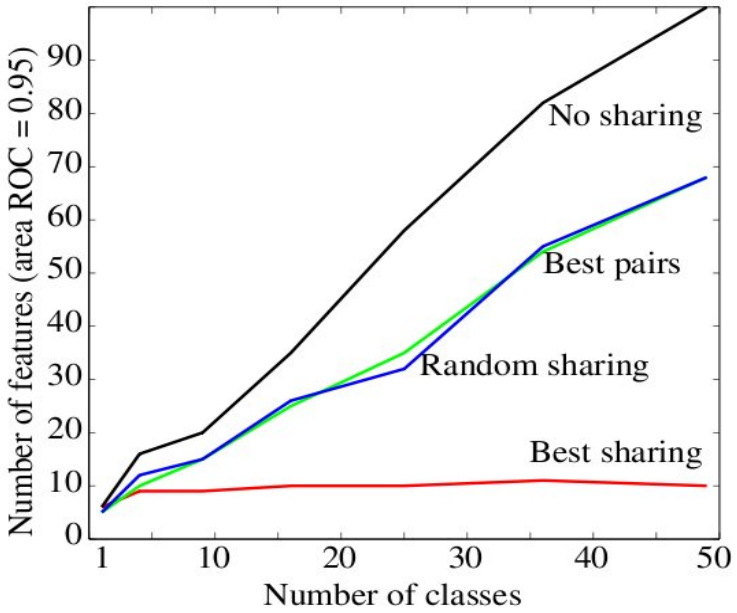
- Algorithm is based on Gentleboost is used
- Each round the best weak learner is added to the strong learners

When?

1. Select the class that has the best reduction of errors
2. The same, but for the second class jointly with the first
3. Until all classes are added
4. Select the sharing that provides the largest error reduction

Comparison


Number of features (area ROC = 0.95)
0
10
20
30
40
50
60
70
80
Best sharing
Best first search
Best pairs
Random sharing
No sharing


Number of features (area ROC = 0.95)
0
10
20
30
40
50
60
70
80
90
1
10
20
30
40
50
Number of classes
No sharing
Best pairs
Random sharing
Best sharing

Sharing features within

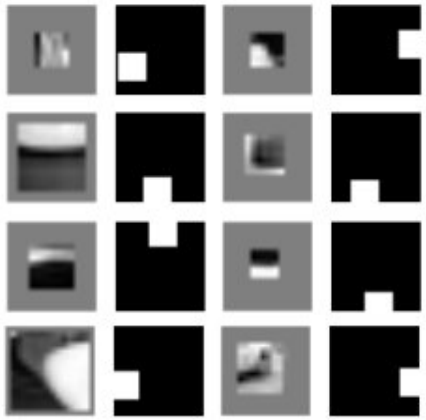
- Multi-class object detection:
  shared features over different types of objects
- Multi-view object detection:
  several points of views of an object may share common visual appearances.
- Location and scale invariant object detection:
  scanning the image and try to find the target

Dictionary of features

Build from:

- Randomly extracted 'patches' from images
- The image here is reduced to 32x32 (as an extraction)
- The patch is 4x4 to 14x14

Dictionary of features



Features

- Approximated by lineair combination of filters
- Patches are extracted from the image
- On average 196 operations per pixel required for 14x14

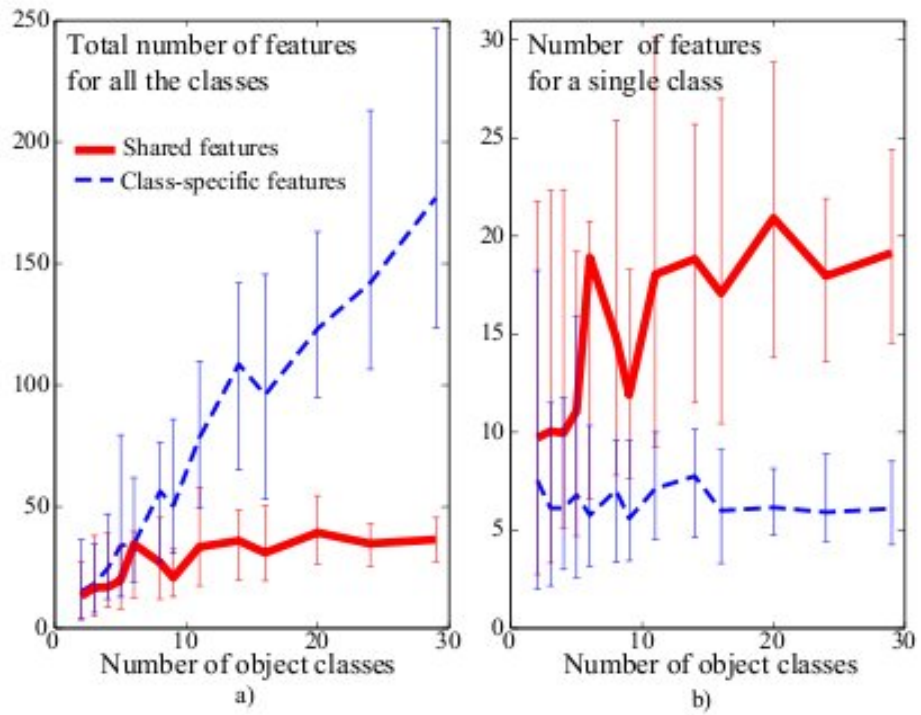
Specific vs Generic features

- Jointly; generalisation across multiple classes
- Independent; specific features for a class

Computational cost

- Jointly; Logarithmic / Sub-linear
- Independent; Linear

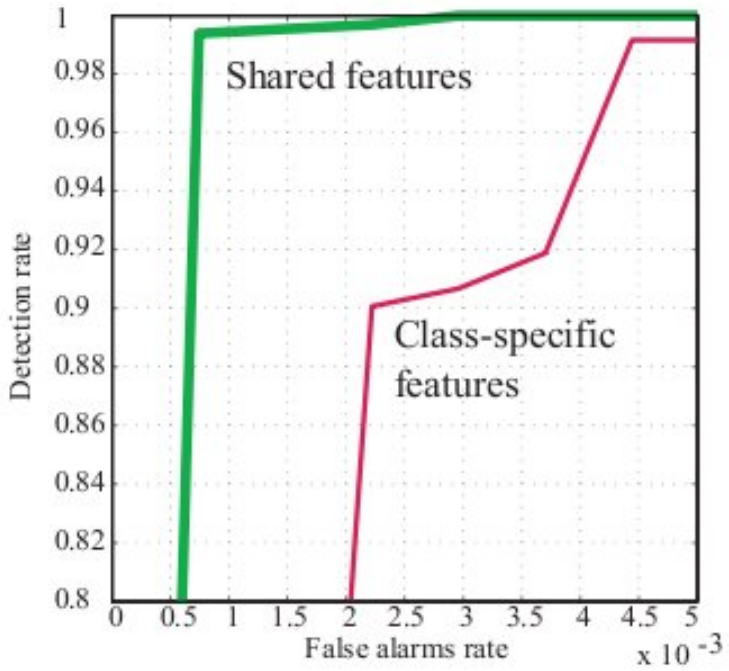
(No) Sharing


Total number of features
for all the classes
Shared features
Class-specific features
Number of object classes
a)
Number of features
for a single class
Number of object classes
b)

Multiview Object Detection

- Large variability
- Overlap

Face detection



Feature sharing across scale/location

Conclusion

- Joint Boosting shared features
- Runs faster (less features)
- Less data to train
- Performance grows sub-linearly
- Better than standard boosting
- Potentially scales up to large number of objects
- Read the extended edition!

Discussion

Discussion