Face Recognition with Support Vector Machines

Nicolas Höning

July 13, 2006

global vs component-based approach

- face recognition techniques
- revision: SVM classifiers
- the global approaches
- the component-based approach
- experimental results

efficient face detection

- the idea
- the algorithm
- results

global vs component-based approach

- **Face Recognition with Support Vector Machines: Global vs Component-based approach**
- a paper by Heisele, Ho and Poggio (MIT)
- compares two approaches to face recognition

global vs component-based approach

- face recognizers became quite efficient through the last years
- but many still rely heavily on stable conditions like pose or illumination (the picture for my new Reisepass had to be taken again because I didn't pose in the exact right way)
- the authors focused on pose invariance
- every approach has a face detection stage and a face recognition stage

binary SVM classifiers

- perform pattern recognition between two classes
- an optimal separating hyperplane is searched
- it is defined by the (support) vectors that lie next to it
- by using kernel functions we can map non-linear separable data to a high-dimensional "feature" space to make it separable

multi-class SVM classifiers

- What if we have more than two (say: q) classes ?
- one-vs-all approach: each SVM separates a single class from all others.
- pairwise approach: each SVM separates between a pair of classes. they get organized in a tree structure (each node is a SVM).
- since there is (yet) no known performance difference, the one-vs-all approach is favored, since the pairwise approach requires q^2 SVMs to be trained.

the global approach

- technically: a single vector represents the whole face image
- global features of the face are mapped

face detection stage

- has as output a normalized picture with only the face part in it
- assures brightness and scale invariance:
- scale: an algorithm moving a window over the picture
- brightness: grey values were normalized between 0 and 1

face detection: examples

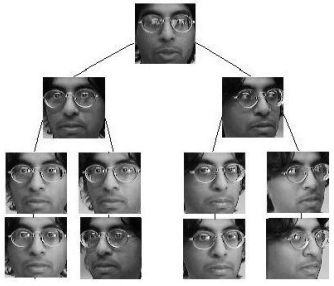
the two global approaches

- first approach: one-vs-all SVMs:

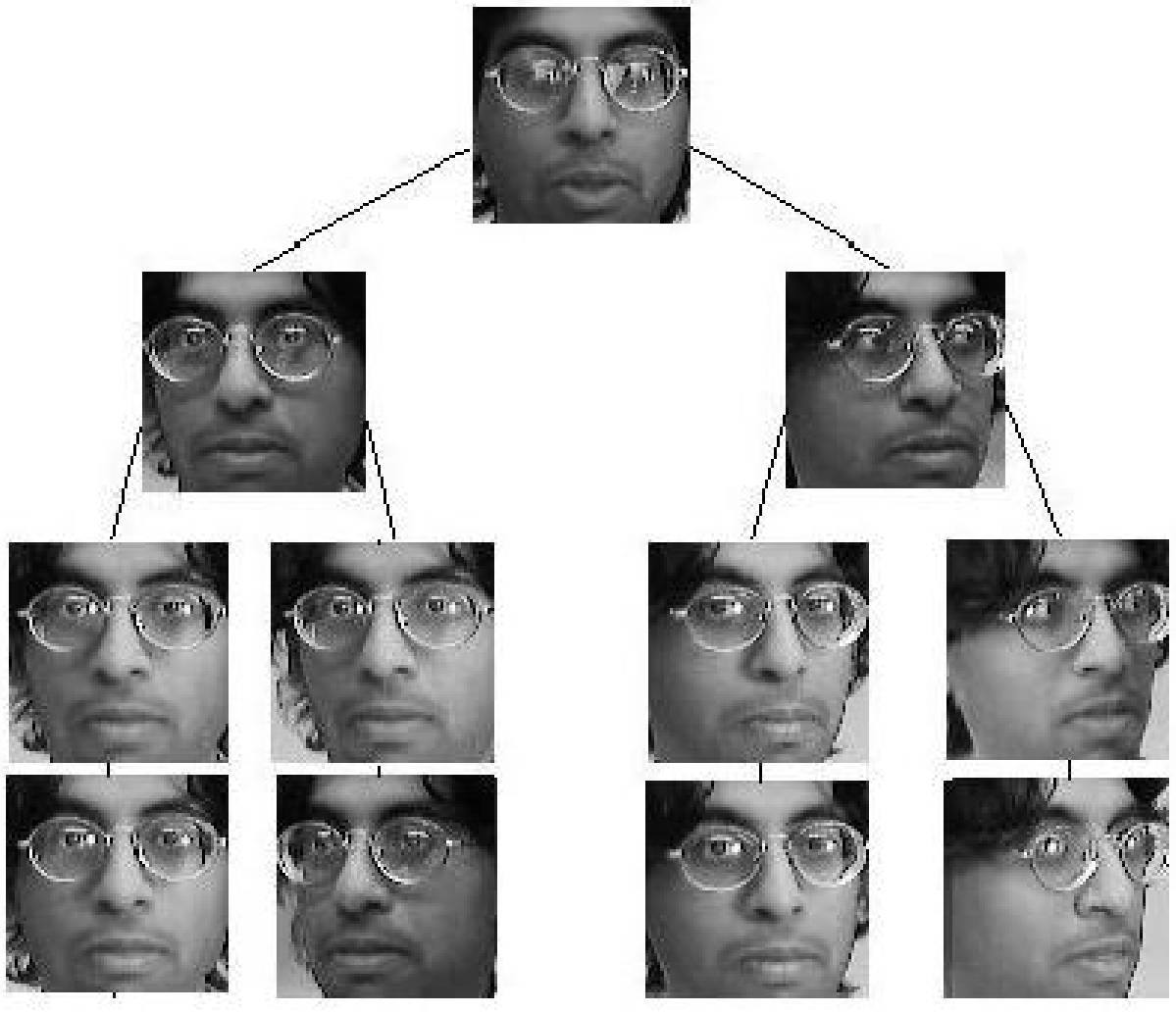
$$y = \begin{cases} n & \text{if } d_n(\mathbf{x}) + t > 0 \\ 0 & \text{if } d_n(\mathbf{x}) + t \leq 0 \end{cases}$$

$$\text{with } d_n(\mathbf{x}) = \max\{d_i(\mathbf{x})\}_{i=1}^{q}$$

- not very robust against pose change, so..
- second approach: first perform a divise clustering stage and train SVM on classifying between clusters
- that leads to a view-point specific tree where average faces are the nodes:

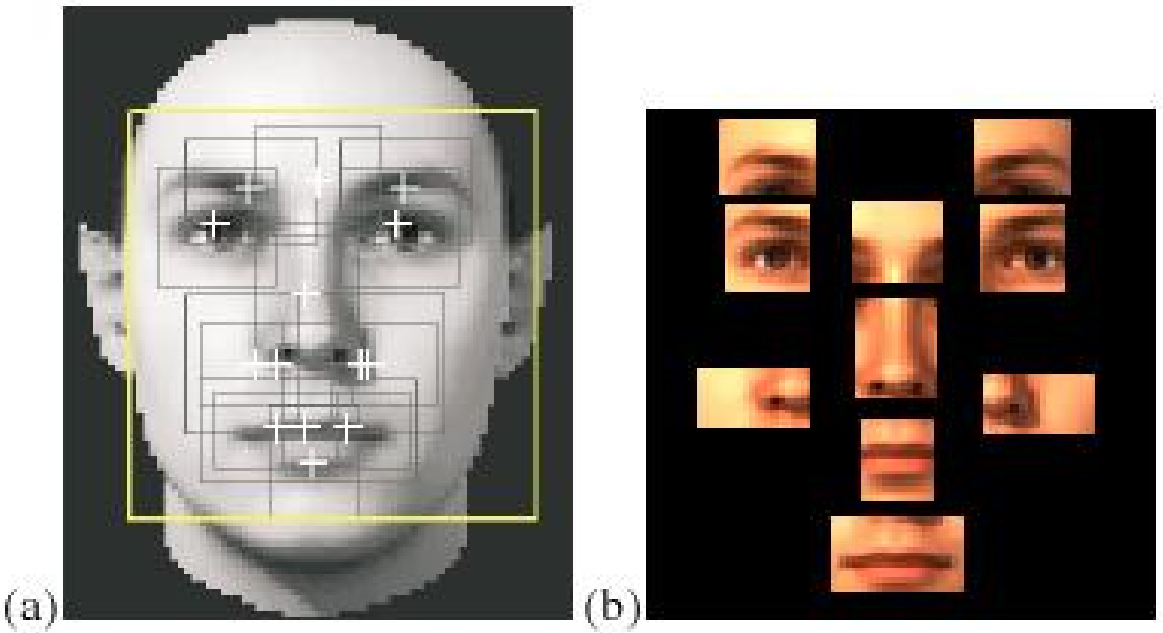


clustering tree



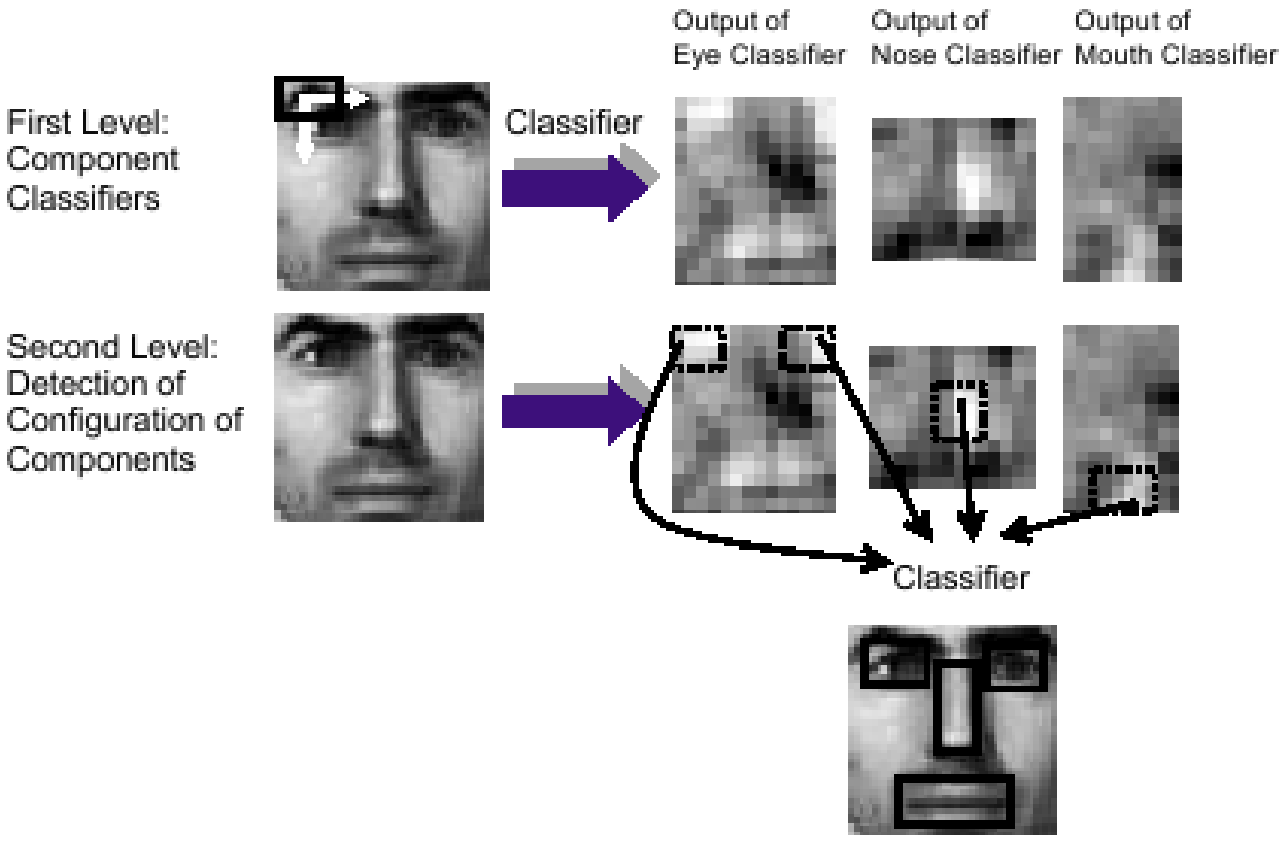
the component-based approach

- this approach only learns parts of the faces (components)



- when the face rotates, the changes within those components are small compared to the global features of the whole face.

face detection



- stage zero: normalize picture as for global approaches
- stage one: detect facial components
- stage two: combines the result of stage one

face recognition

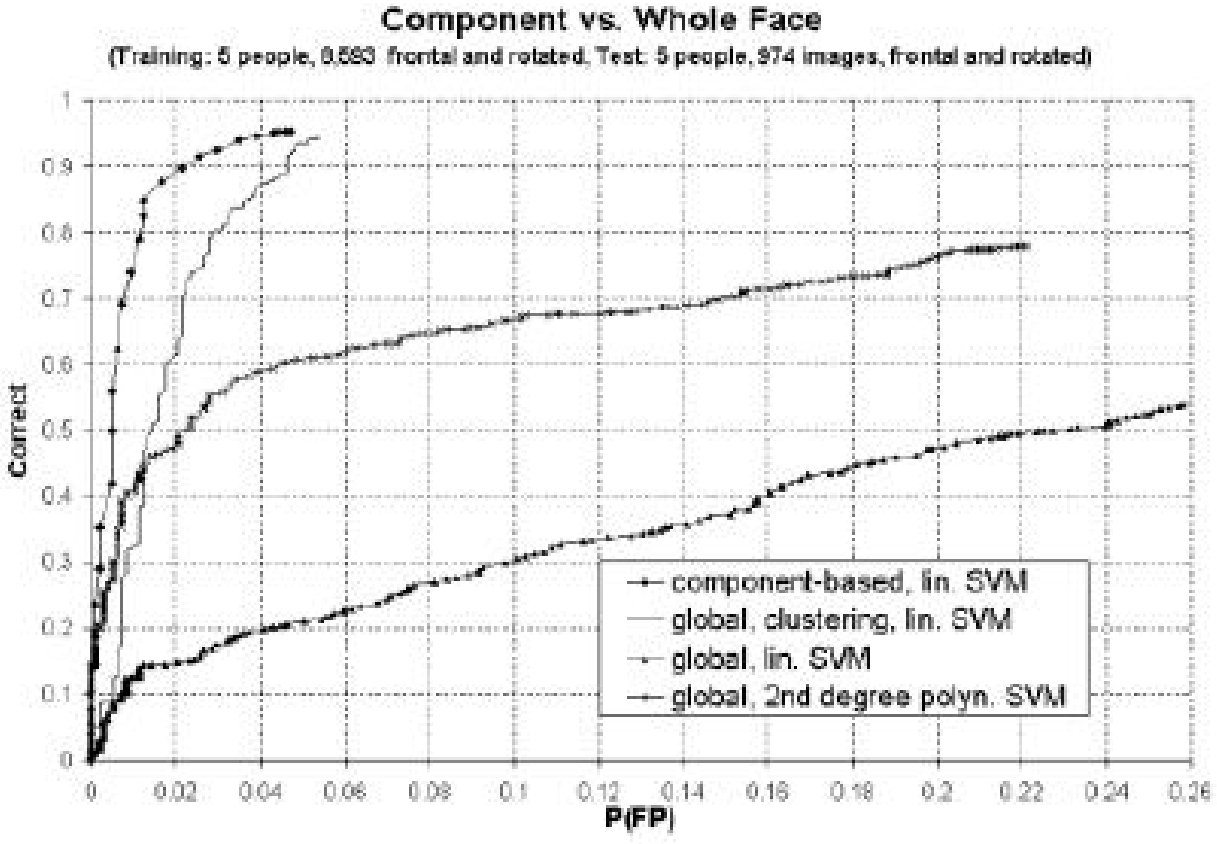
- components get normalized (size and grey-values)
- use one-vs-all SVM approach again (one person = one class)

experimental results

- two experiments were done
- training set: 8,593 grey face images of five subjects (1,383 frontal views).
- between 80x80 to 130x130 pixels
- rotation up to 40^o
- the test data was 974 pictures with different illumination and background

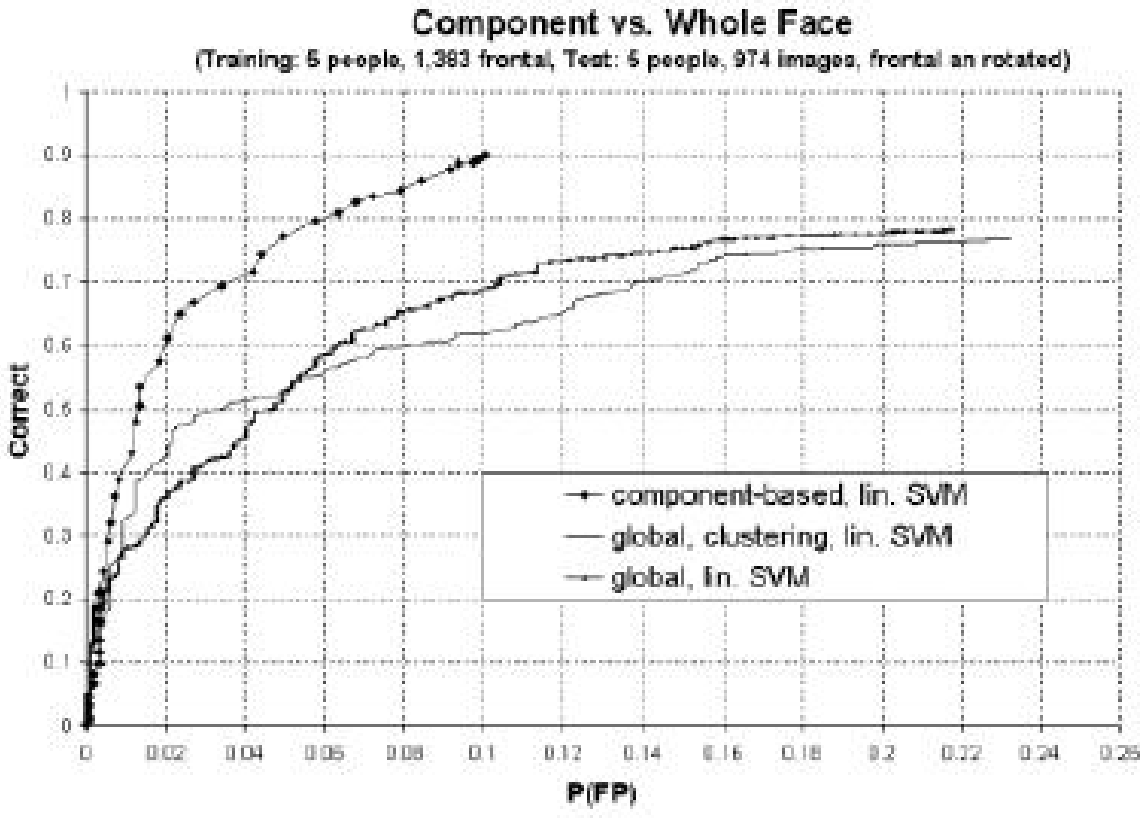
Experiment 1

all pictures in training and test set



Experiment 2

only frontal view pictures were used in the training set



conclusion

- the component system was always the best method even though it had less powerful SVMs
- clustering in the global approach led to a significant improvement, even over non-linear SVMs
- rotation is too complicated for linear global classifier

some results

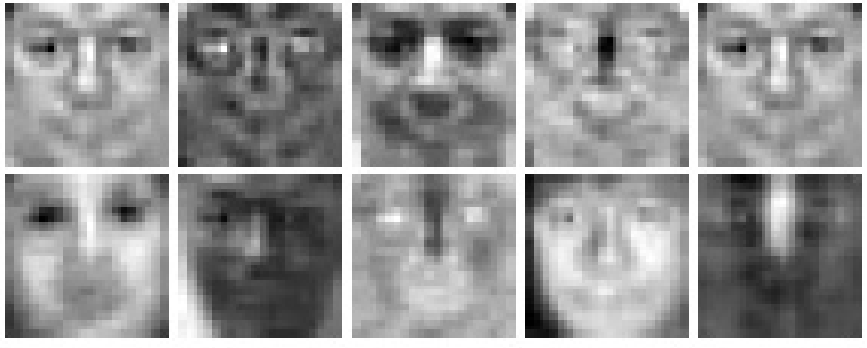
Figure 8. Examples of component-based face recognition. The first 3 rows and the first image in the last row show correct identification. The last two images in the bottom row show misclassifications due to strong rotation and facial expression.

efficient face detection

- **efficient face detection - a paper by Romdhani, Torr, Schölkopf, Blake**
- it deals with the stage of face detection
- and proposes an idea to be far more efficient with that

svms are slow classifiers

- proportional to number of support vectors (i.e. training examples - in face recognition there are quite a lot needed)
- idea: can we compute a small set of vectors out of the set of support vectors so that the classification works almost as well?
- we're looking for something like this:



reduced set vectors

- we have a decision surface: $\Psi = \sum_{i=1}^{N_x} \alpha_i \Phi x_i$
- we want something like this: $\Psi' = \sum_{i=1}^{N_z} \beta_i \Phi z_i$
- where N_z is much smaller than N_x (a lot less vectors involved) and $\Psi - \Psi'$ (the introduced error) gets minimized

minimizing $\Psi - \Psi'$

- the first reduced vector z would have a span (lineare Hülle) of $\Phi(z)$
- we want to minimize the orthogonal projection of Ψ to $\Phi(z)$
- that problem can be reduced and then transformed w.r.t. k (the kernel function)
- when we have Ψ' with m reduced vectors, the (m+1)th vector can be computed from that to yield Ψ''. Ψ'' minimizes the distance to Ψ even further (but takes more computing time to classify, of course)

the SRSM algorithm

now, the whole Sequential Reduced Set Machine algorithm they propose works like this:

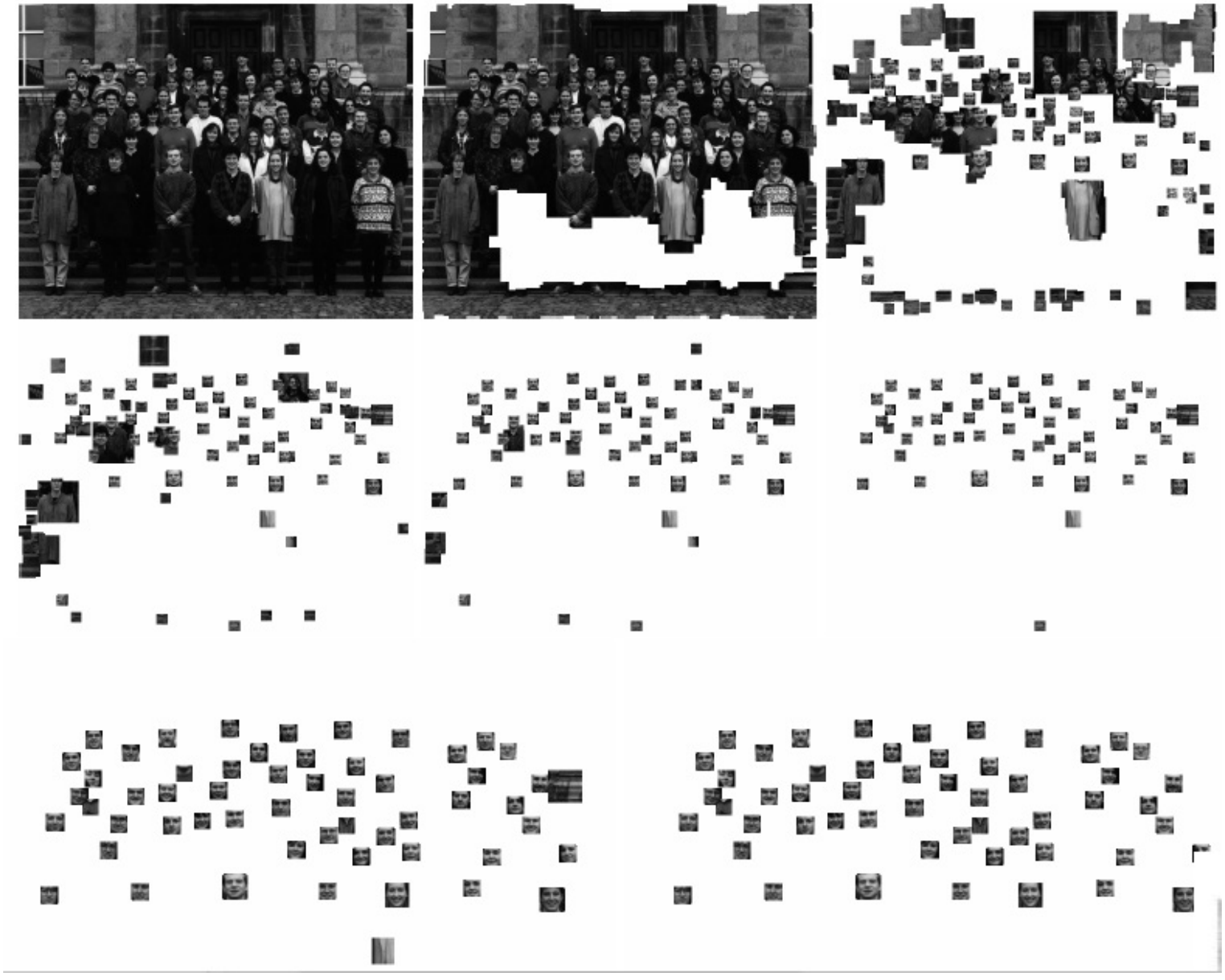
- 1. start with the first of the reduced set vectors (m=1)
- 2. evaluate the given patch
- 3. if the result is smaller zero, we can reject the patch and stop
  if not, increment m, try step 2 again
- 4. if all of our reduced set vectors have been used and the result is still $>= 0$, try with the whole SVM.

- the idea is: there are many, many patches that could be a face (from a pixel to the whole image). the huge majority of them can be thrown out by very few reduced set vectors (that have been calculated in advance).

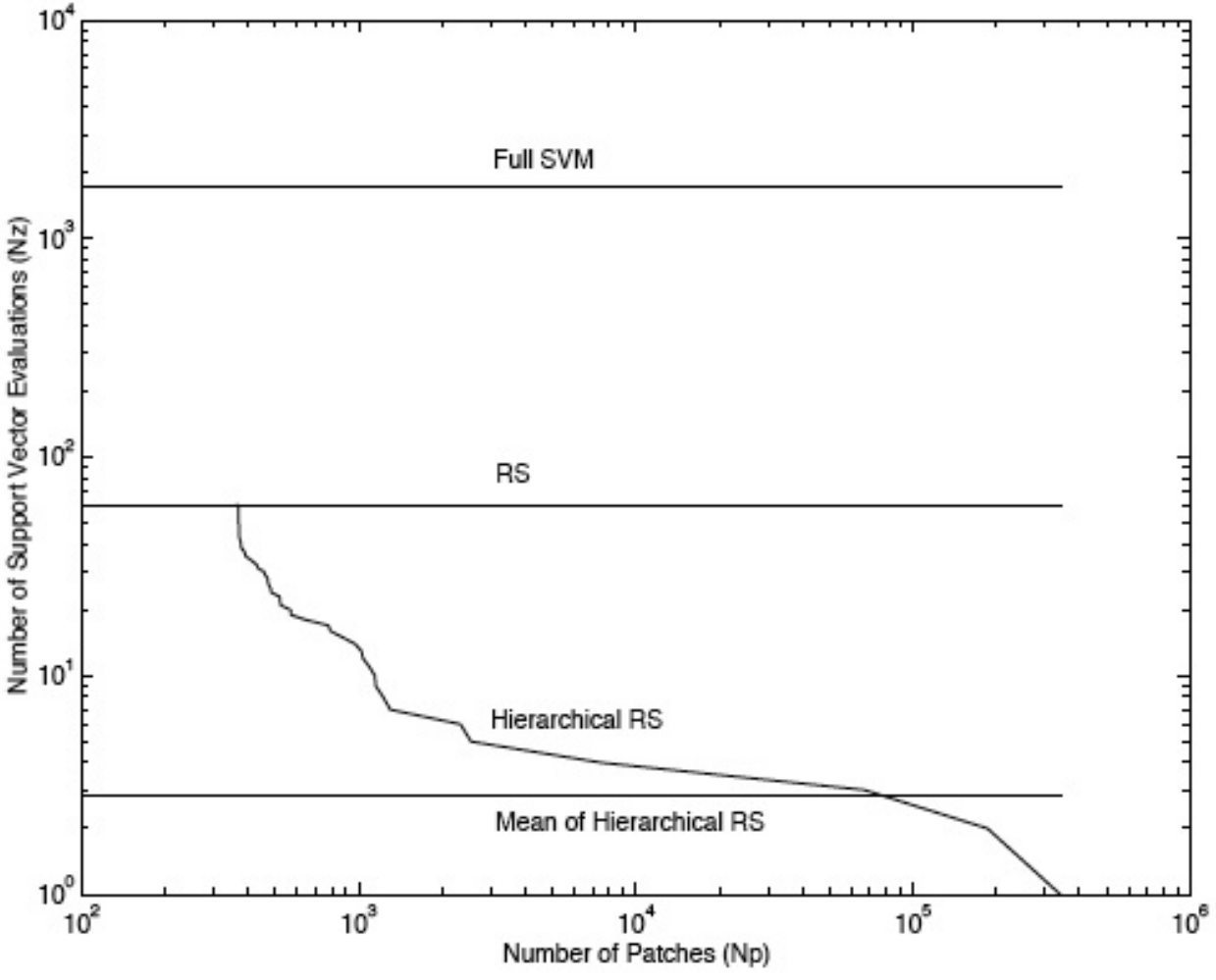
results

let's have a quick glance at the results they achieved:

a big family photo



comparison



the algorithm

- speed improvement: the SRSM system is 30 times faster than the RSM system
- accuracy: they used a test set that was being used by other researchers and did slightly worse (comparison is still hard because they did no preprocessing in this experiment)
- idea 1: they use the Gaussian Kernel as distance metric, something else might even be more suitable
- idea 2: if we found a face of size x, shouldn't we then prefer patches of sizes similar to x?
- this method can also be applied to other problems than face detection

the end

questions?