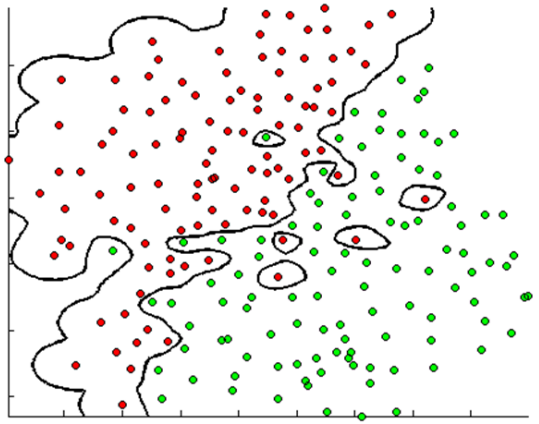
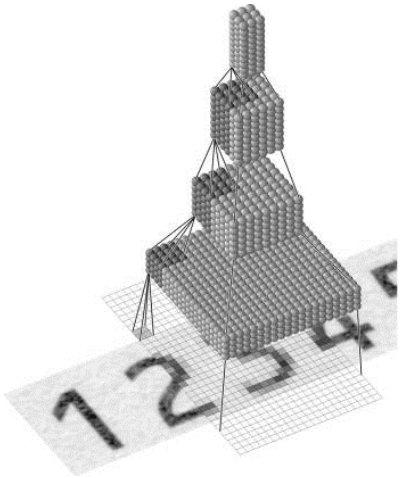
Vision algorithms on embedded platforms

Virginie Moser, CSEM, **Vision embedded systems**

Swissphotonics workshop, Neuchâtel, 4th February 2016





Why embedded vision ?

- Infinite number of applications

- Decision making
  - Recognition
- To be everywhere, vision needs to be :
  - Low cost
  - Low power
  - Small / light
- Fast algorithms

Embedded vision at CSEM

- VIP: Vision In Package
  - Smallest vision system able to run a standalone vision application

- Algorithms and demonstrators
  - People and car detection (CNN)
  - Gesture recognition (Boosting)

VIP: Vision in package

- Smallest vision system
  - Able to run a standalone vision application

- Hardware:
  - Imager Aptina MT9V024 (752x480)
  - Cortex M4F 180-MHz
  - 64MBytes of SDRam

- Operating system:
  - uKOS RTOS

Supervised learning (CNN, Boosting)

- Subset of machine learning based algorithm
  - Need labelled database
    - Images with their label (sample)

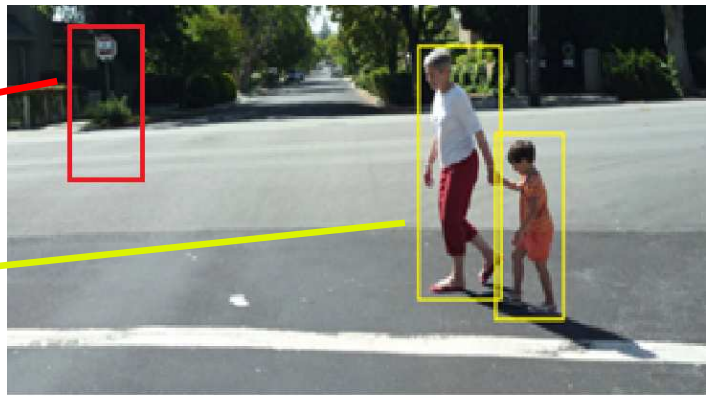
- Optical character recognition

0 1 2 3 4 5 6 7 8 9 Negative

- People detection

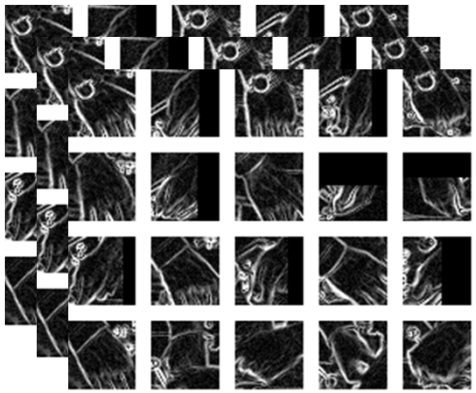
Negative sample

Positive samples



Supervised learning principle

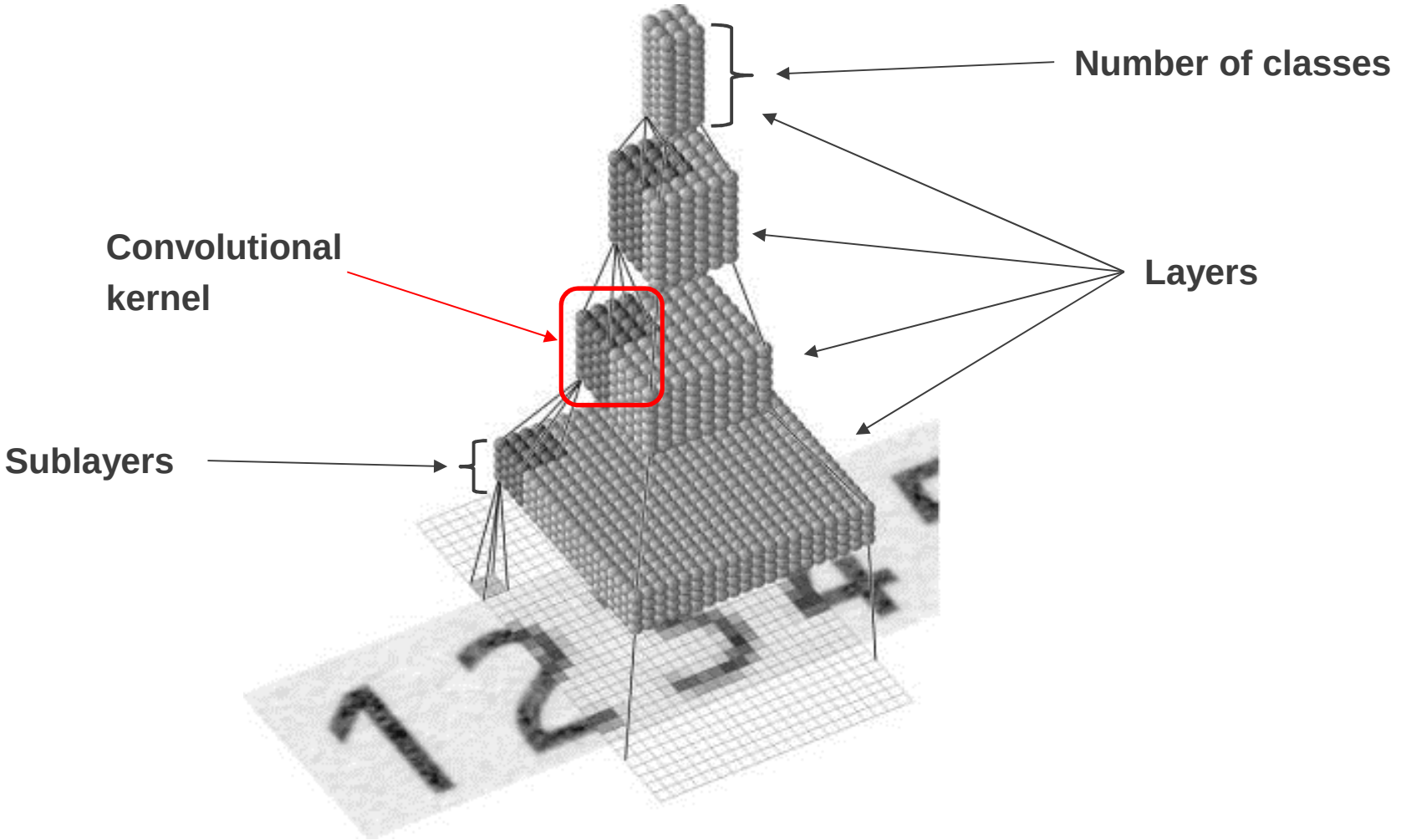
Off-line **Training**



On-line
Execution

Answer: “hand, or no hand”

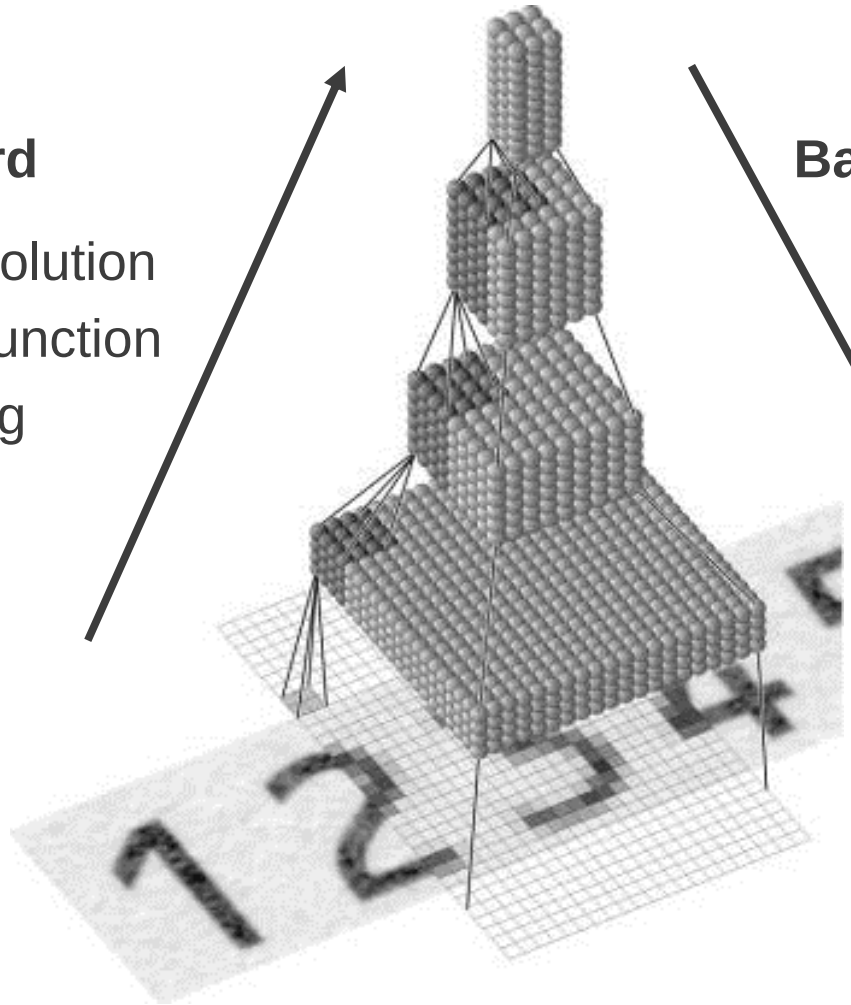
Convolutional Neural Network (CNN)



CNN Training phase (off-line)

Feedforward

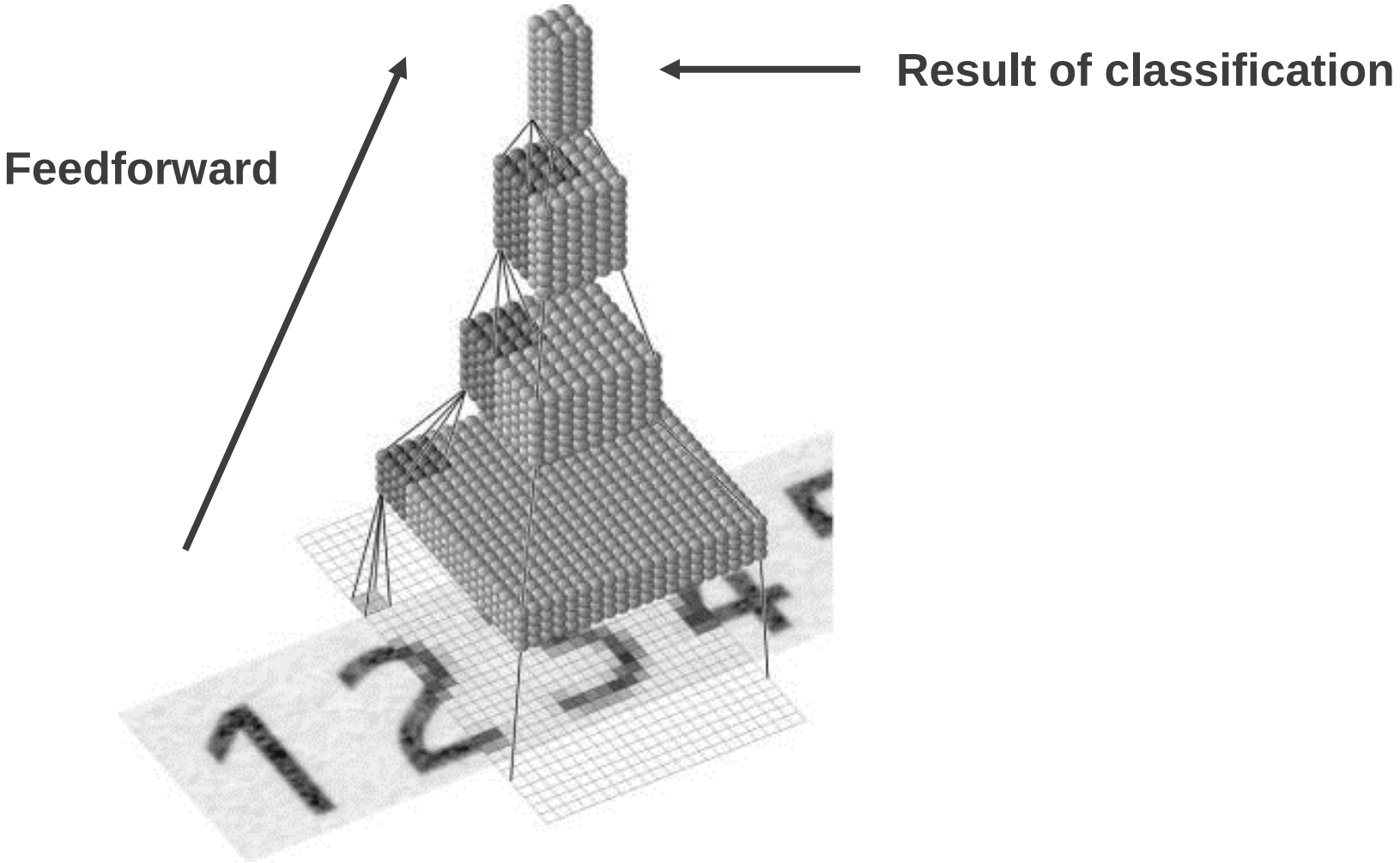
Kernel convolution
Non linear function
Subsampling



Backpropagation

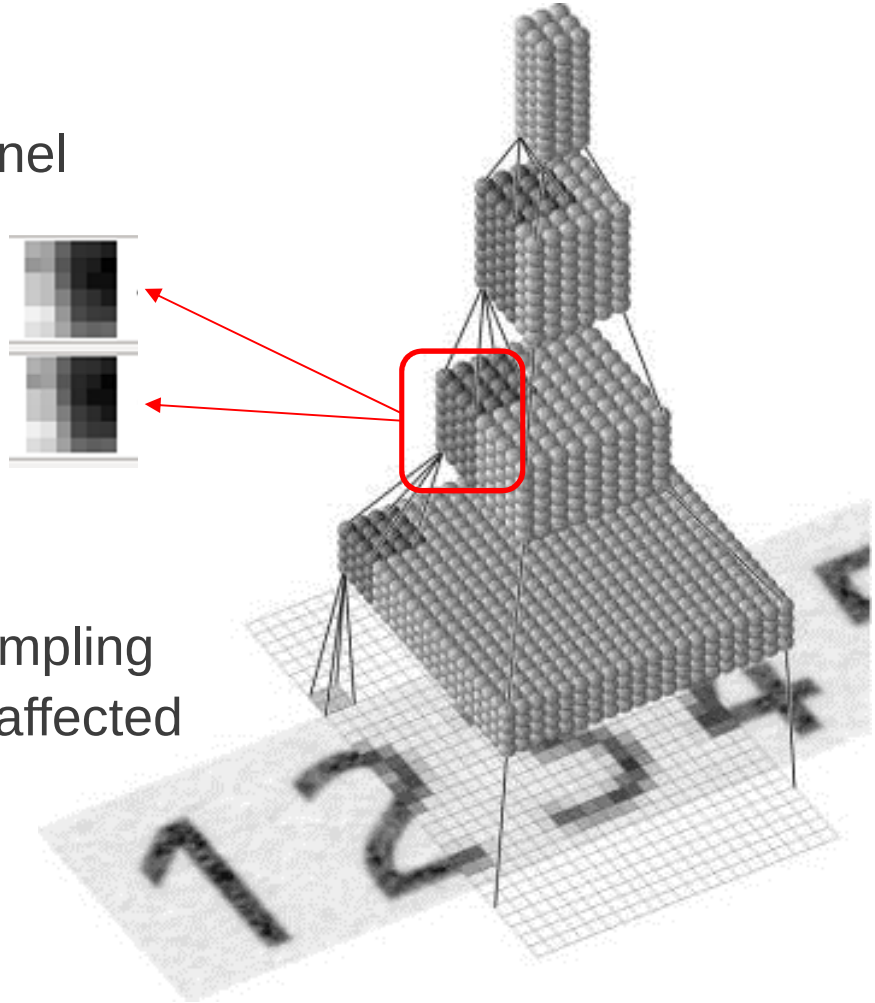
Compute error
Backpropagate error
Update kernels weights

CNN Execution phase (on-line)



CNN speed optimization

- Number of layers and sub-layers
  - Visualization of computed kernel weights
  - Remove useless sublayers

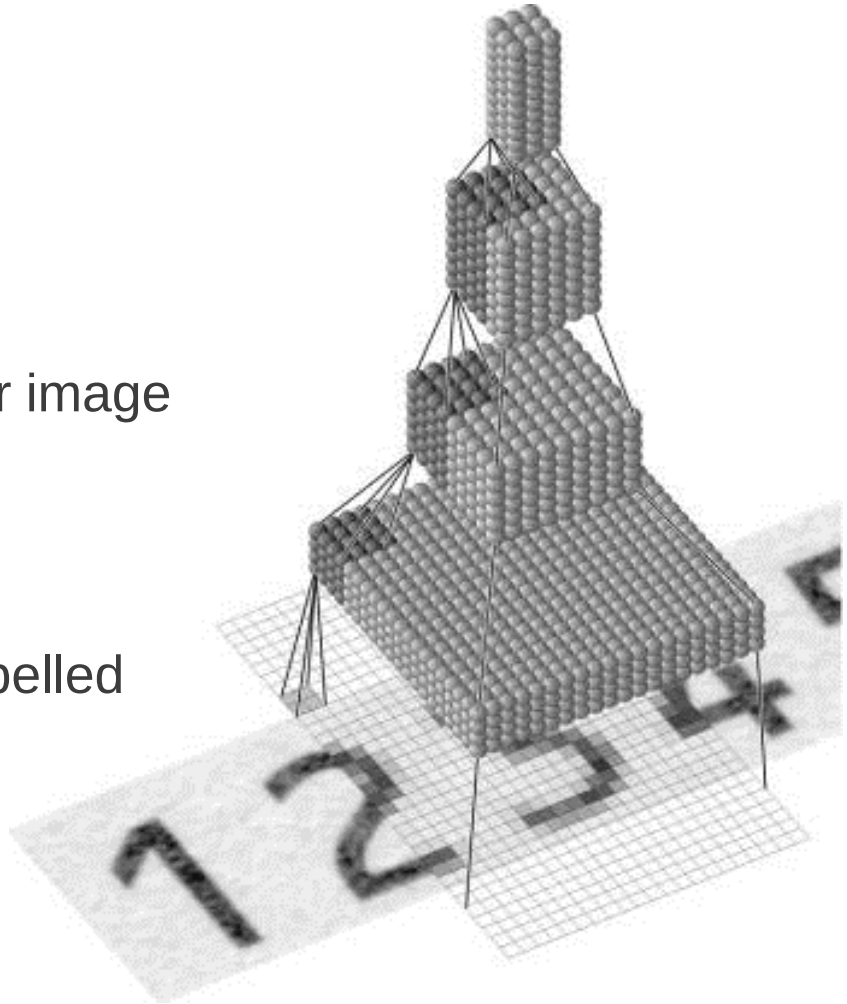


- Implementation optimization
  - Merge convolution and subsampling
  - Recognition rate only slightly affected

CNN : Advantages / drawbacks

- Advantages
  - Fast
  - Very good reliability
  - Few memory usage
  - Can detect the object on bigger image

- Drawbacks (off-line)
  - Need a lot of representative labelled data
  - Training phase

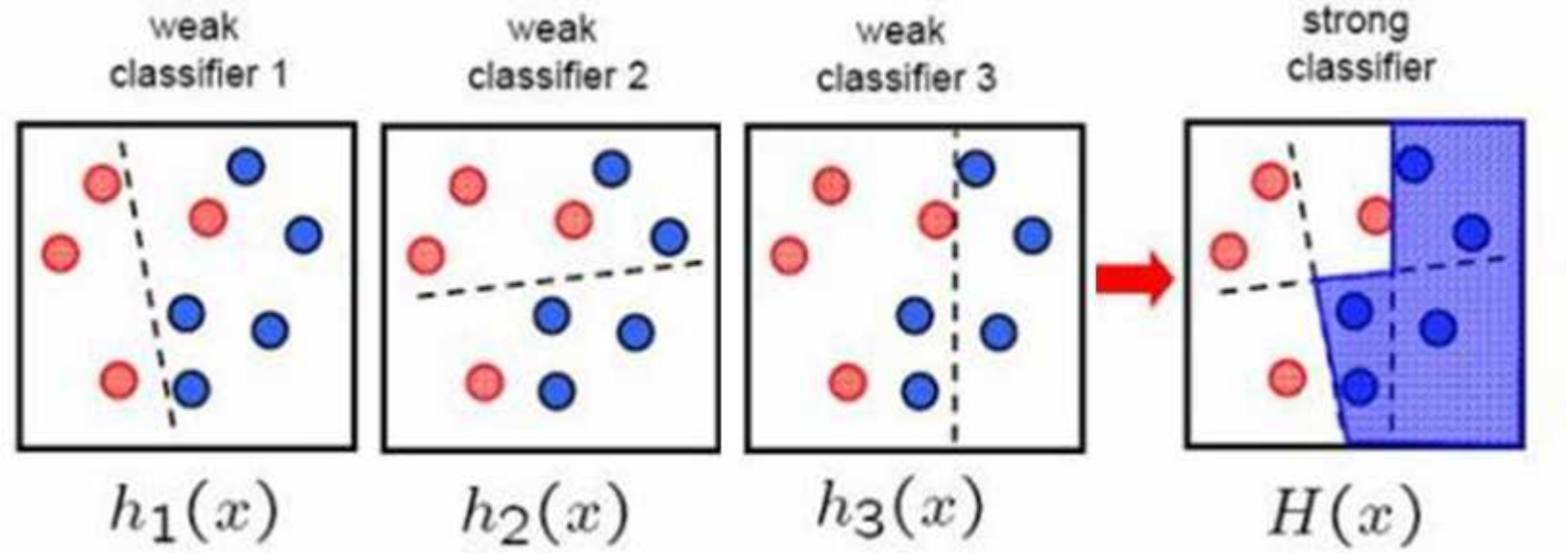


Boosting algorithm : Training principle

- Based on user-defined features
  - For example Haar features

- Algorithm computes the best combination of features
  → Weak classifier (~8 features)
- One weak classifier computed at each iteration
- At each iteration, focuses on misclassified samples
- Strong classifier
  - Combination of all weak classifiers (few hundreds)

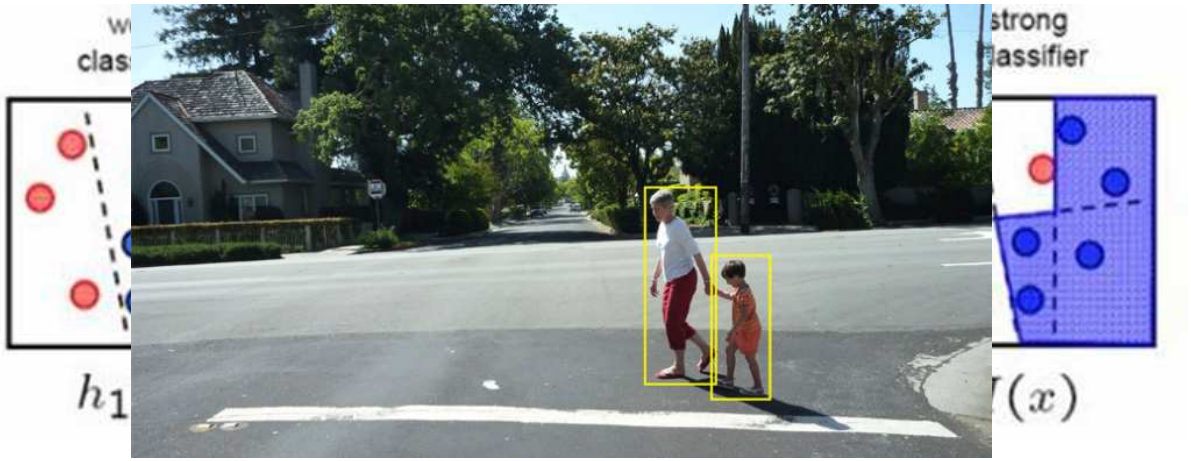
Boosting algorithm: illustration



Cascade of classifiers

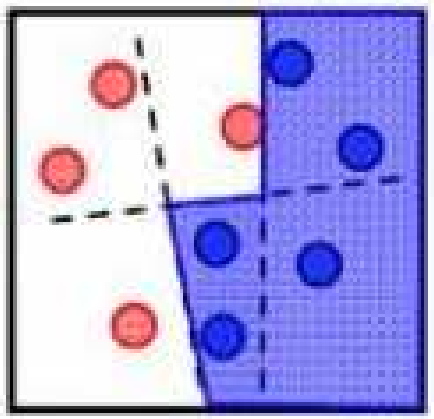
Boosting: execution optimization

- At execution, the whole cascade does not need to be executed
  - Decision can be made after first weak classifier



  - Important speed up especially for real world images where the majority of the image does not contain the searched object

Boosting : Advantages / drawbacks



- Advantages
  - Extremely fast
  - Good reliability
  - Few memory usage

- Drawbacks (off-line)
  - Need a lot of representative labelled data
  - Training phase

Extremely fast algorithm with very good reliability

- Use Boosting to perform a quick detection
  - Determine Region of Interest (ROI)
- Use CNN to affine the recognition

- The labelling burden can be reduced by:
  - Using existing databases
  - Developing a tool for auto-labelling
  - Use the availability of negative samples
  - Add noise and deformation to positive samples

Demonstrator: People and car detection

- Manage traffic lights at cross-roads
  - Detect pedestrians (adults, children but not animals)
  - Detect cars (all types, buses, but not pedestrians nor animals)

- Model with Playmobils, toys car

Demonstrator: People and car detection

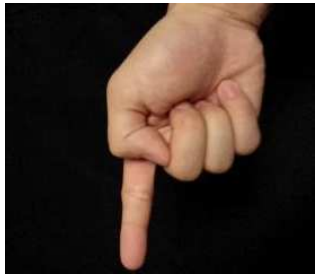
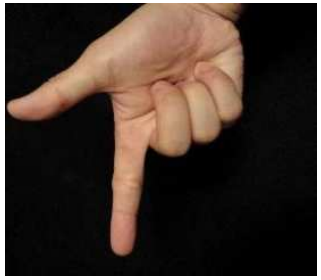
- Performances of CNN on VIP
  - Car detection (120x120 pixels): ½ second
  - Pedestrian detection (200x100 pixels): ½ second
  - Memory usage: 65 Kbytes
  - Labelling needs for recognition rate > 99%
    - 700 images for pedestrians
    - 250 images for cars

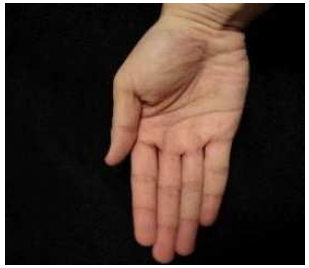
Video: people and car detection

Demonstrator: Gesture recognition

- Classify 6 different hand gestures with Boosting



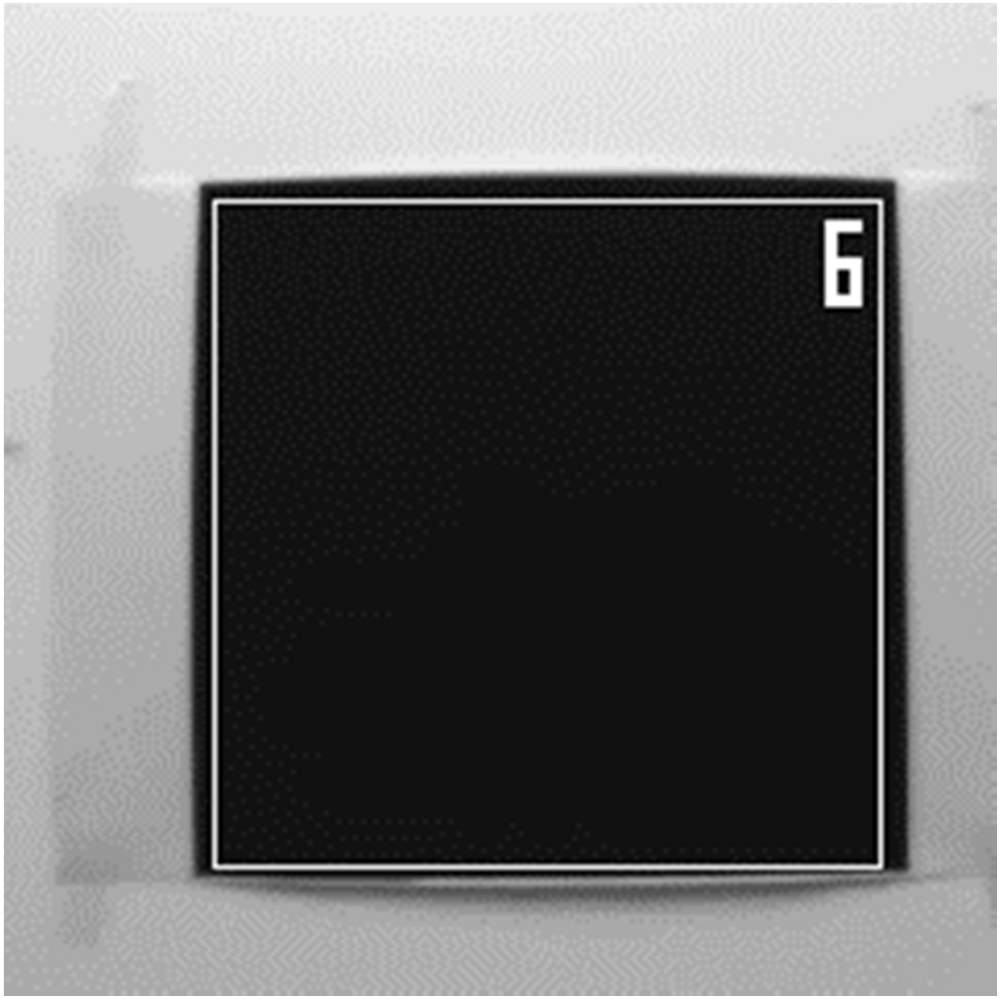


The dataset is taken from the work of Song et al., "In-air Gestures Around Unmodified Mobile Devices", 2014.

Demonstrator: Gesture recognition

- Performance of Boosting on VIP
  - Gesture classification: 100ms
  - Memory usage: 3 Mbytes (2’000 weak classifiers)
  - Labelling needs for recognition rate > 99%
    - 16’000 images

Video: Gesture recognition



Questions ?

Come and Play !

See our demonstrator in Show Room !