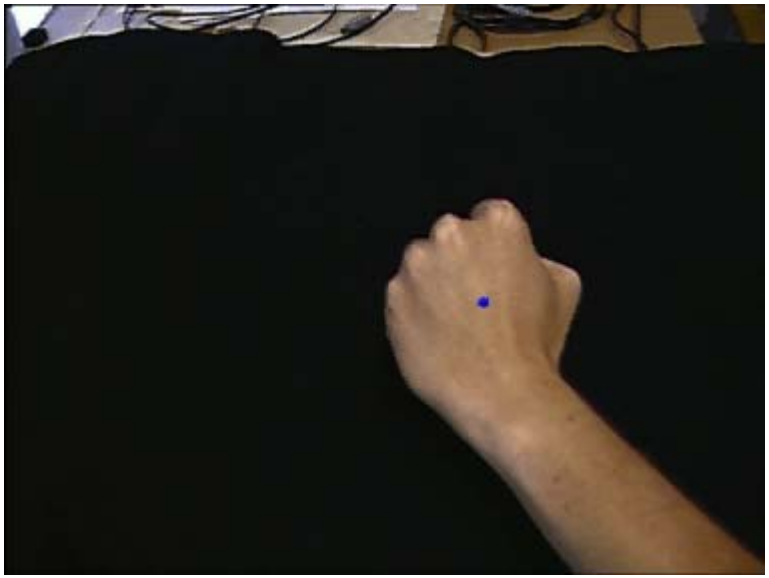
Segmentation in video

Advanced Image Processing

VGIS8 – Spring 2009

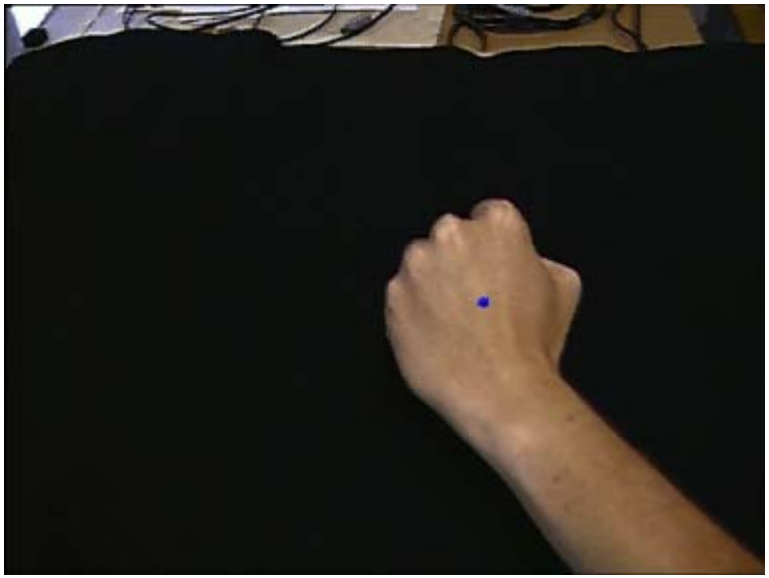
Segmentation in video

Basic idea: detect objects of interest



Segmentation in video

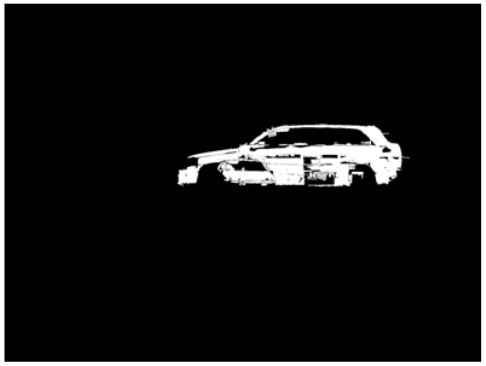
Basic idea: detect objects of interest



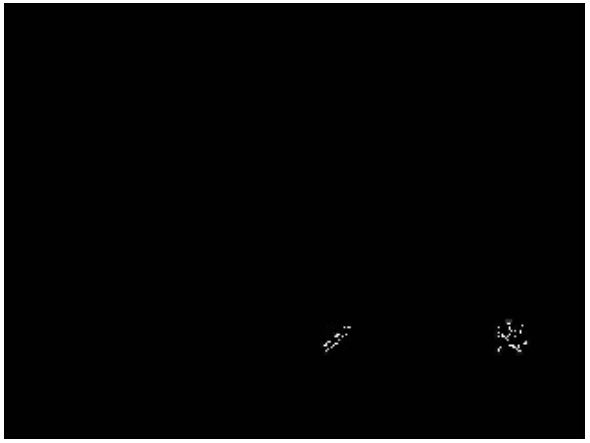
Result: binary image

General algorithm

1. Save reference image
2. Capture current image
3. Perform image subtraction
4. Thresholding
5. Filter noise



Results - General algorithm



Today's topics

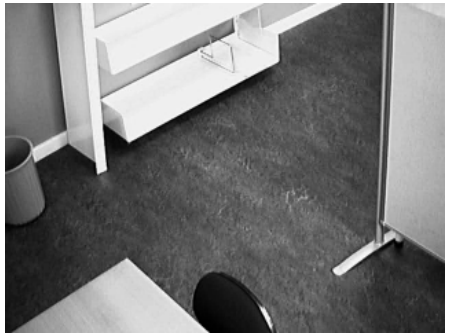
- Background subtraction
- Constructing the reference image
- Updating the reference image
- Global and local thresholds
- Image differencing
- Noise filtering
- Advanced segmentation

1. Acquire reference image
2. Capture current image
3. Perform image subtraction
4. Thresholding
5. Filter noise

Background subtraction

1. Acquire reference image
2. Capture current image
3. Perform image subtraction
4. Thresholding
5. Filter noise

- First frame as reference image

$r(x,y)$: 

$f(x,y)$: 

- Pixel-wise subtraction

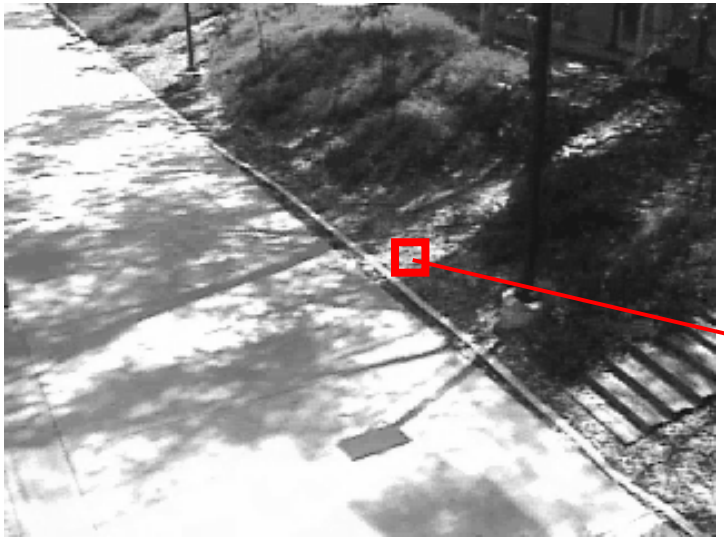
$$g(x,y) = f(x,y) - r(x,y)$$

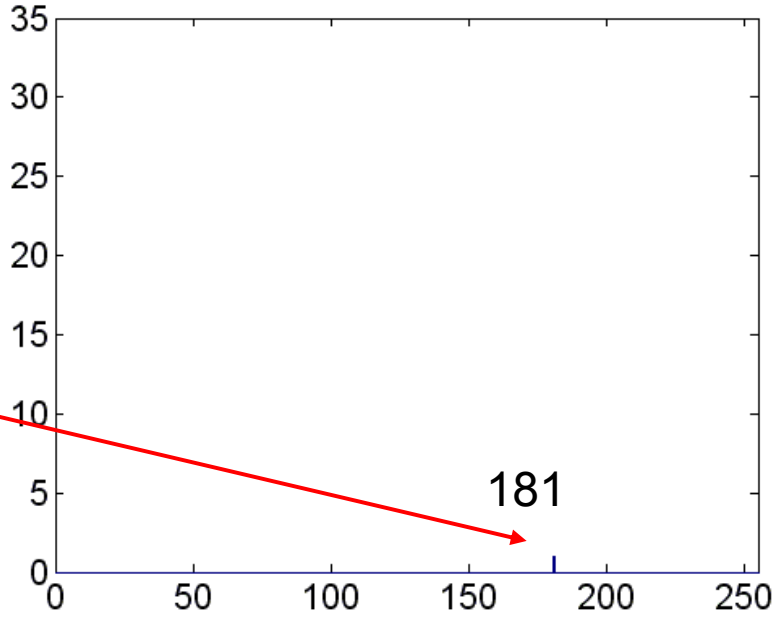
- Apply absolute value

$$g(x,y) = |f(x,y) - r(x,y)|$$

Constructing the reference image

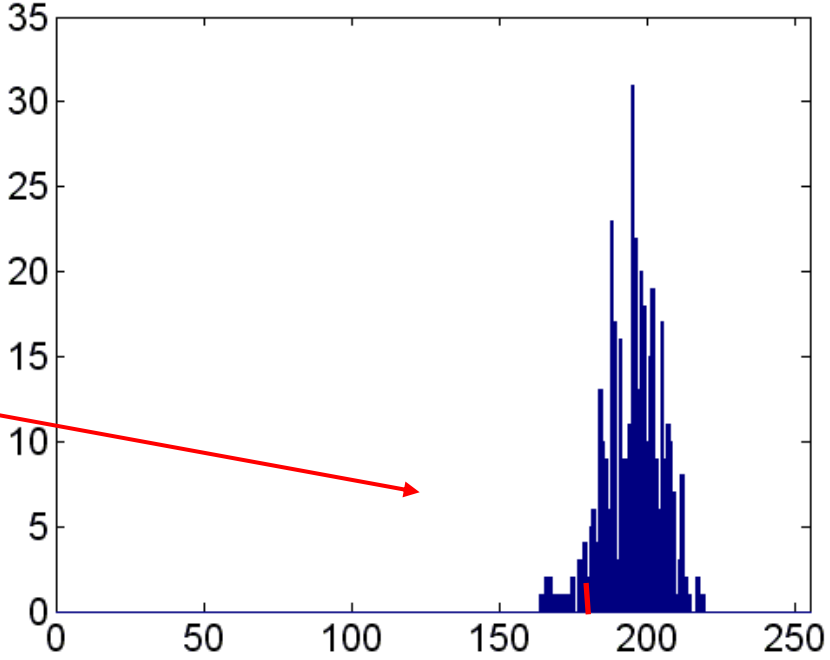
- With non-stable background
  - e.g. camera noise, changes in light, small motion from wind gusts
- Then learn the background





Constructing the reference image

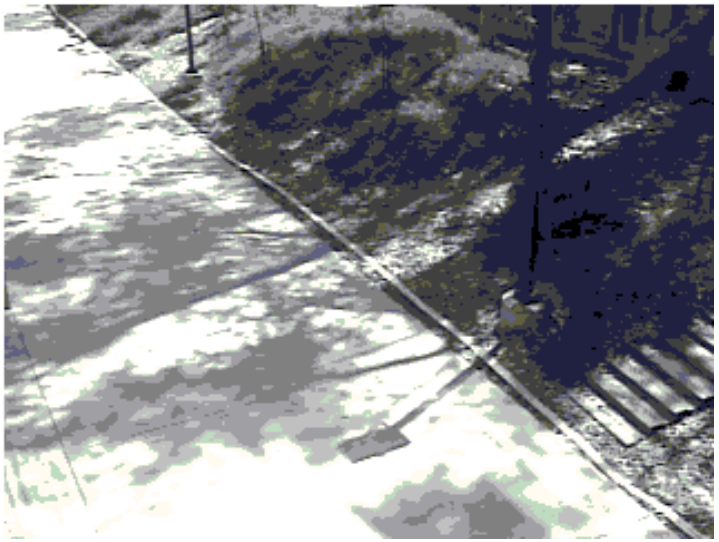
- With non-stable background
  - e.g. camera noise, changes in light, small motion from wind gusts
- Then learn the background



Constructing the reference image

- Learning the background
  - Capture *N* empty images and calculate the average background image

$$r(x,y) = \sum_{i=1:N} f_i(x,y) \,/\, N$$



1. **Calculate average reference image**
2. Capture current image
3. Perform image subtraction
4. Thresholding
5. Filter noise

Updating the reference image

- With gradual changes during processing
  - e.g. light changes during a day
- Then update the reference image

Updating the reference image

- With gradual changes during processing
  - e.g. light changes during a day
- Then update the reference image
  - Update speed α

$$r_{new}(x,y) = \alpha \cdot r_{old}(x,y) + (1- \alpha) \cdot f(x,y)$$

Update speed
$\alpha = 0.98$

Global vs. local thresholds

1. Acquire reference image
2. Capture current image
3. Perform image subtraction
4. Thresholding
5. Filter noise

- So far:
  - Image subtraction

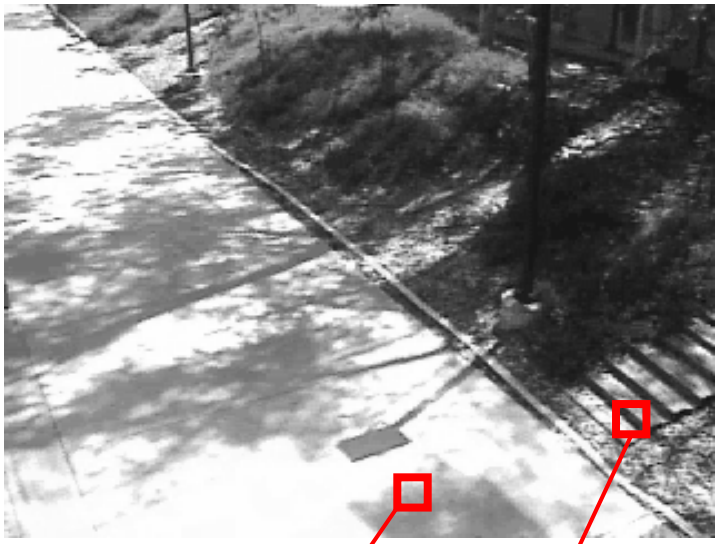
$$g(x,y) = |f(x,y) - r(x,y)|$$

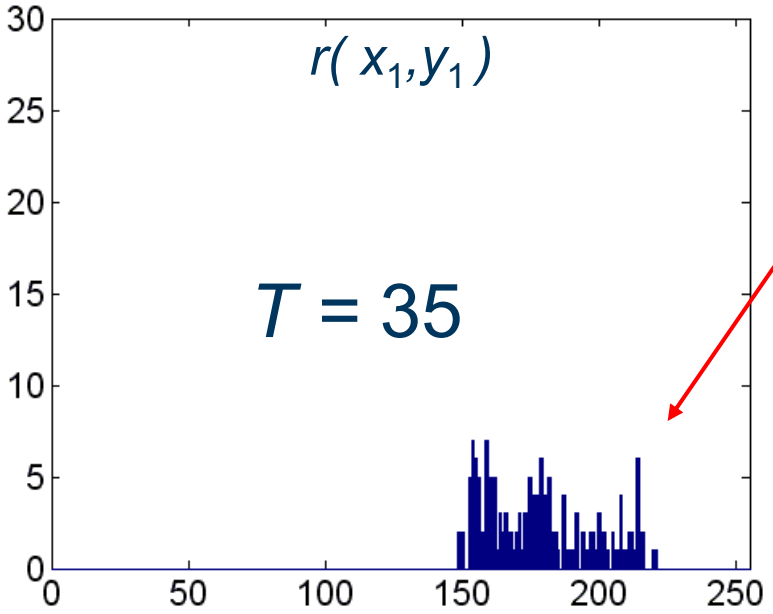
  - Global threshold T to obtain binary image b

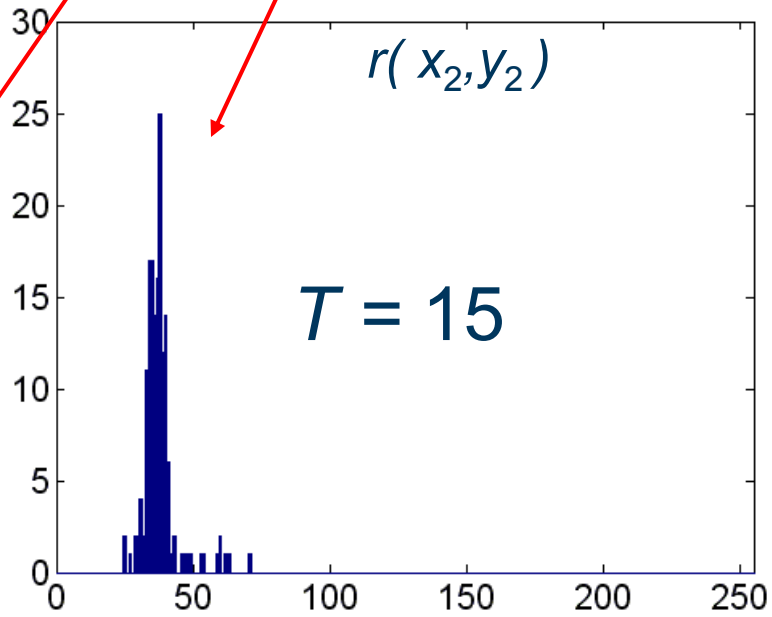
$$b(x,y) = \begin{cases} 0, & \text{if } g(x,y) < T \\ 255, & \text{else} \end{cases}$$

Global vs. local thresholds

- With local pixel variation
- Then apply local thresholds





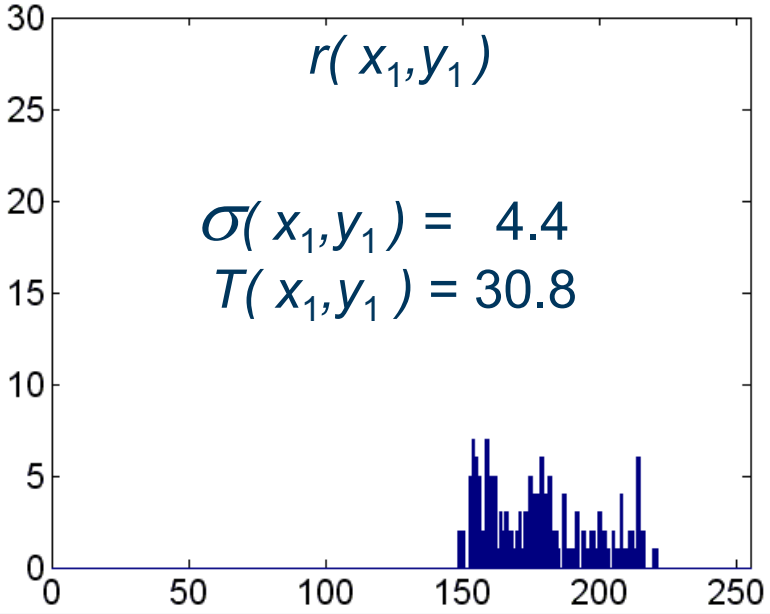


Global vs. local thresholds

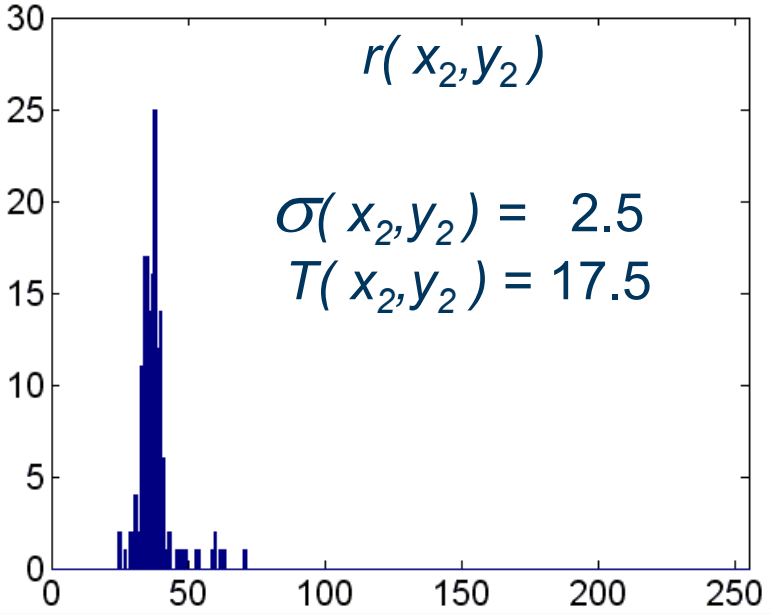
- Apply local threshold
  - Calculate mean and standard variation σ
  - Define T in terms of σ

$$T(x,y) = \beta \cdot \sigma(x,y)$$

  - *Choose (global) β*



$\beta = 7$



Global vs. local thresholds

- Apply local threshold

1. **Calculate mean and standard variation for each pixel**
2. Capture current image
3. Perform image subtraction
4. **Threshold according to standard variation**
5. Filter noise

- Local thresholds + reference image updating is a powerful background subtraction method – especially if you can choose your background

Image differencing

- With very dynamic scenes which can not be learned
- Then choose the previous frame as the reference frame

Image differencing

- With very dynamic scenes which can not be learned
- Then choose the previous frame as the reference frame

$$r_t(x,y) = f_{(t-1)}(x,y)$$

- Different reference image (depending on application)

$$r_t(x,y) = f_{(t-k)}(x,y)\ ,\ k = 2,3,\ldots$$

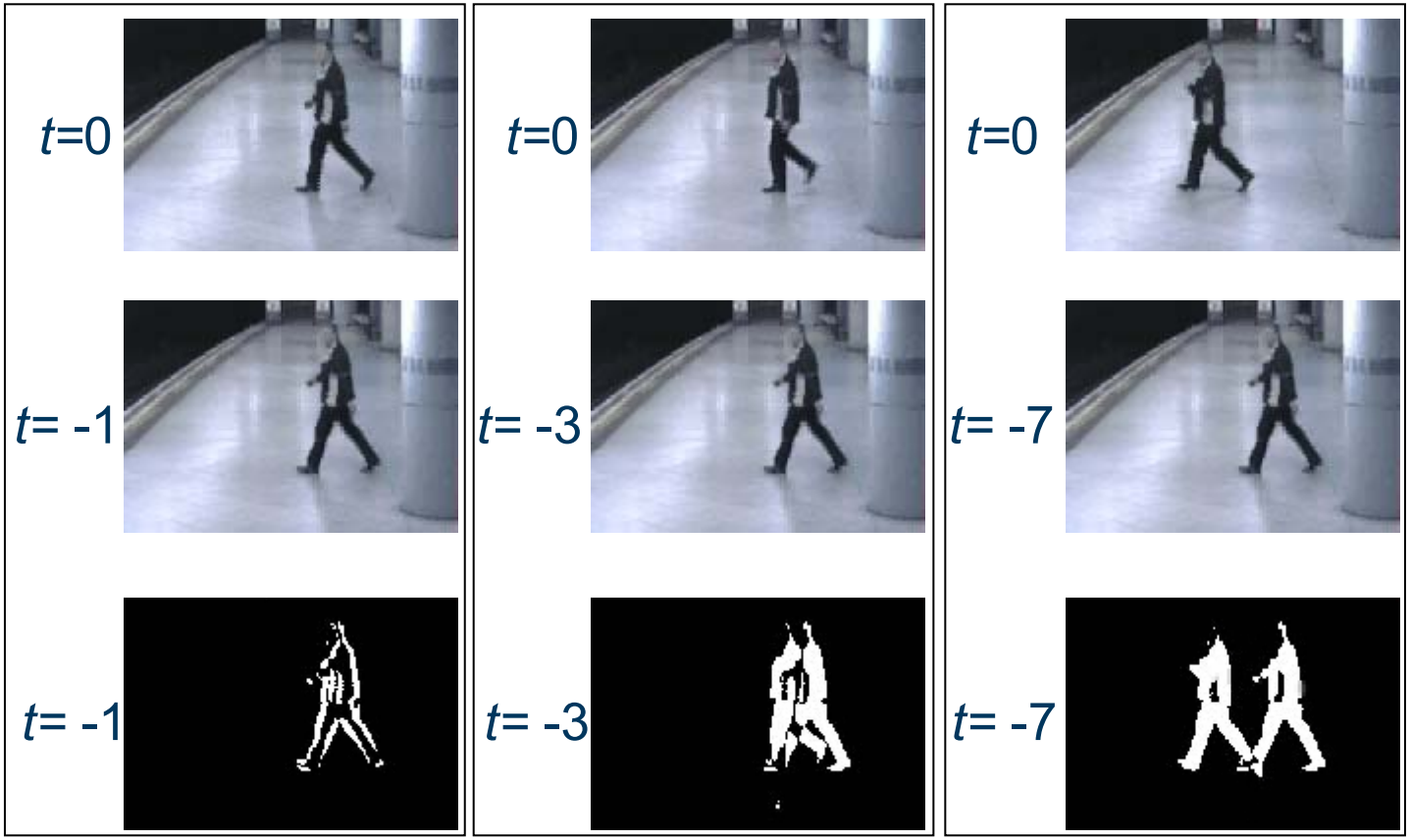
- Double differencing

$$g_t(x,y) = [f_t(x,y) - r_{1,t}(x,y)] \text{ AND } [f_t(x,y) - r_{2,t}(x,y)]$$

Image differencing vs. background subtraction

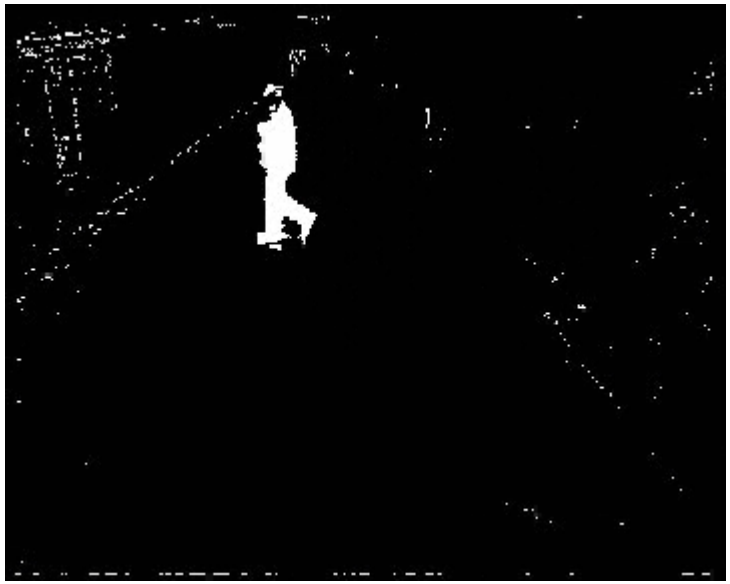
Image differencing	Background subtraction
• Handles dynamic scenes • Detects only motion • Detection results in ghosts	• Requires static background • Detects stationary objects • Detects whole objects

Ghosts:



Noise filtering

- Single pixels are likely to appear
  - Pixel-noise
- Apply Median filter
  - Depending on filter size, bigger spots can be erased
- Alternatives:
  - Morphology
  - Temporal filtering
  - Restrictions on size, shape



Summing up…

- The general algorithm for segmentation:

1. Acquire reference image
2. Capture current image
3. Perform image subtraction
4. Thresholding
5. Filter noise

- Learn reference image from a set a frames
- Update the reference image continuously
- Learn the standard variation and apply local thresholds
- Apply image differencing in dynamic scenes

Next topics – advanced segmentation

- Shadows and light changes
- Multi-modal background models
- Learning background with non-empty images
- Layered background models
- Optical flow for segmentation

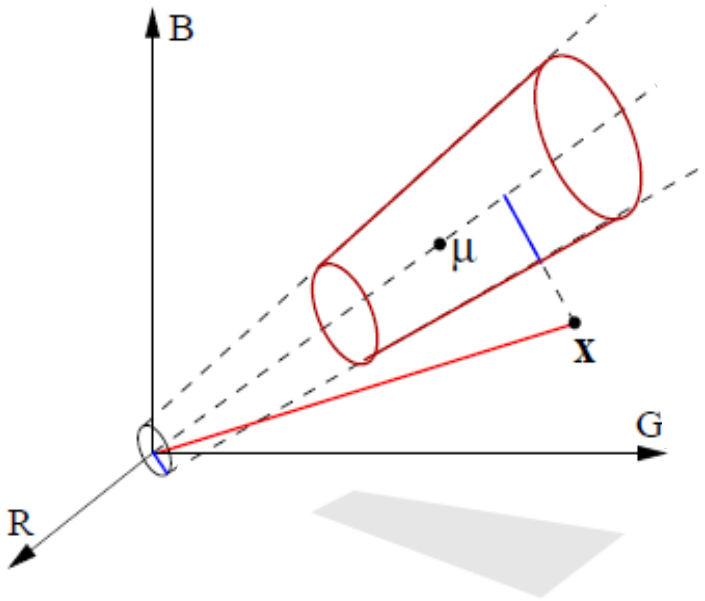
Shadows and light changes

- Shadows and light changes cause false positives

- Solution: separate color and intensity
  - HSV-color space
  - Codebook background subtraction method
    - One method to handle advanced issues in segmentation

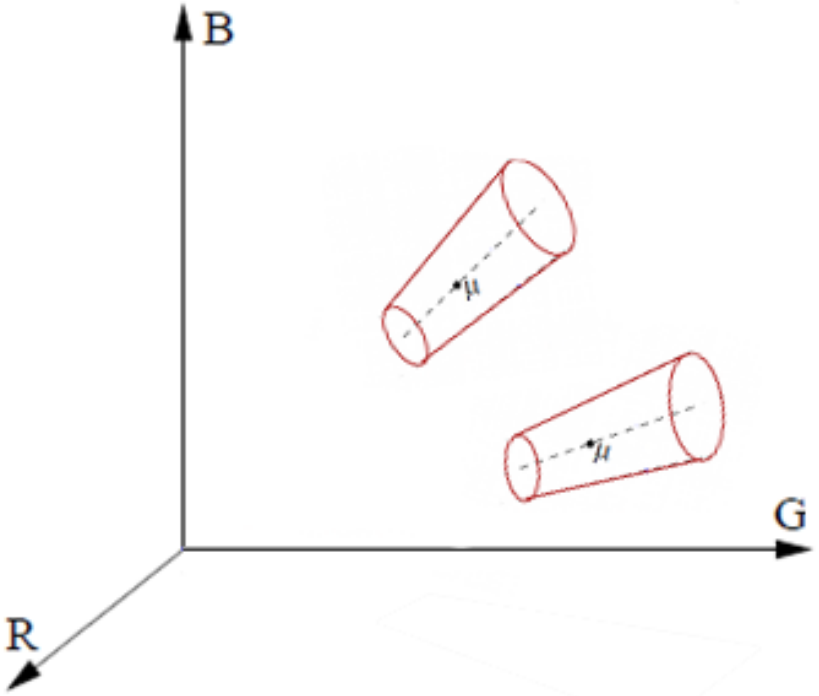
Shadows and light changes

- Color-model of codebook background subtraction:



Multi-modal background models

- Learn the background from *N* frames – *N*~300
- Allow multiple codewords for each pixel



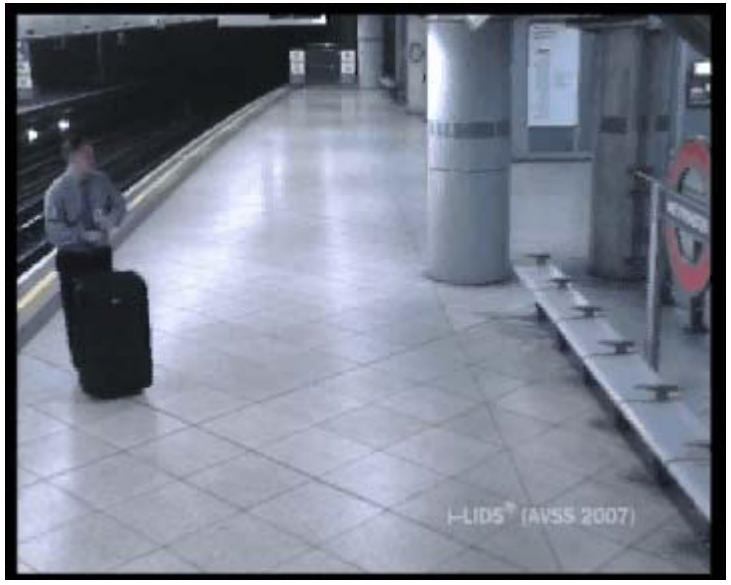
Learning background with non-empty images

- Learn the background from *N* frames – *N*~1000

- Evaluate all codewords after training
  - True background vs. false background
  - True: pixel values appear periodically
  - False: pixel values appear in bursts

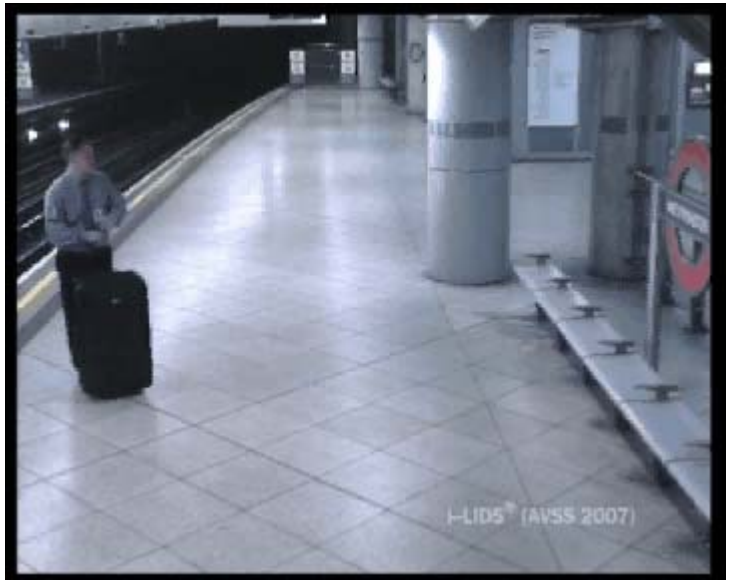
Layered background models

- Parts of the scene can become temporary background
  - e.g. parked cars, left behind luggage, the train
- Reference image updating adapts over time



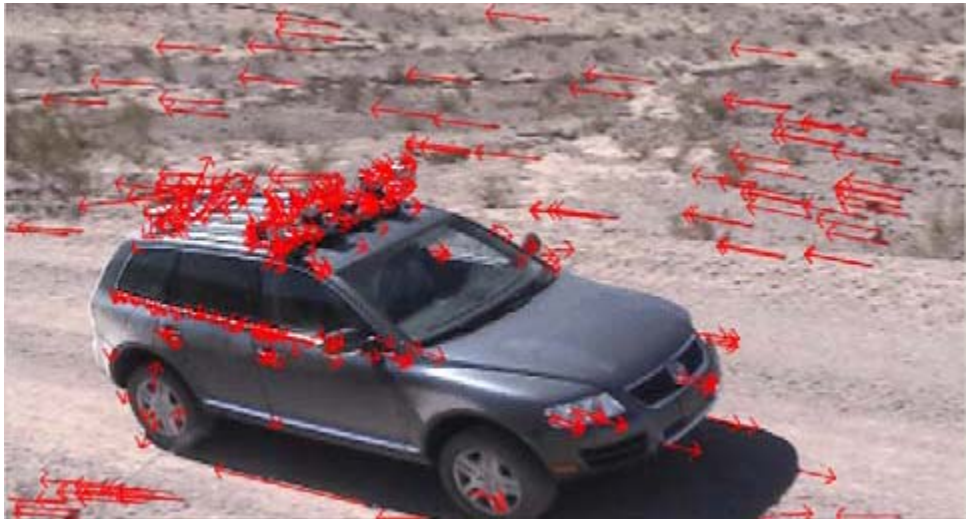
Layered background models

- Alternative: use multiple layers of background
  - Continuously evaluate all foreground regions
    - IF pixel values appear periodically for *N* frames
    - THEN add new codeword to the background model
  - Delete expired codewords



Optical flow for segmentation

- Moving cameras make reference images impossible (e.g. car-mounted cameras)
  - Both objects and background move
- Solution:
  - Estimate motion direction between frames
  - Objects with motion different than average motion can be identified



Summing up …

- Advanced background subtraction addresses many real life problems
- … but the problem of segmentation is not solved
- Still an area of research:
  - Combining color segmentation with edges
  - Applying regional information
  - Applying contextual information

Exercises

1. Acquire reference image
2. Capture current image
3. Perform image subtraction
4. Thresholding
5. Filter noise

- Download Matlab script + videos
  - Segment hand.avi with different thresholds
- Implement:
  - Learn reference image from a set a frames
    - Use the first 100 frames in sun_and_trees.avi
  - Update the reference image continuously
    - Use the first 40 frames of office_light.avi for background
  - Learn the standard variation and apply local thresholds
    - Use the first 100 frames in sun_and_trees.avi
  - Apply image differencing in dynamic scenes
    - sun_and_trees.avi is a good test case
- Mini project

Mini-project

- To do (for each group)
  - Find information and learn the principle
  - Implement a working prototype
  - Hand-in report (3-5 pages) about: principle, usage, algorithm, your implementation and results
  - Present your work (10-20min)
- Hand-in report: 9/4 at 12.00
- Presentation: 13/4