CAP6411
Computer Vision Systems
Lecture 6

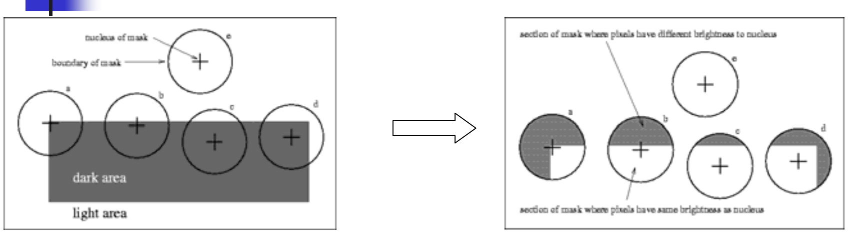
Alper Yilmaz

Office: CSB 250

Email: yilmaz@cs.ucf.edu

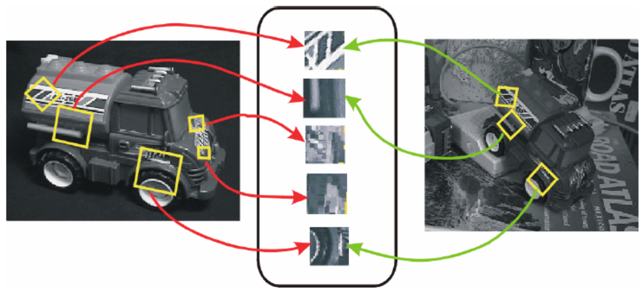
Web: http://www.cs.ucf.edu/courses/**cap6411/cap6411**/spring2006

Recap
The SUSAN Detector



$$R(r_0) = \begin{cases} g - n(r_0) & \text{if } n(r_0) < g \\ 0 & \text{otherwise} \end{cases}$$

Recap
SIFT - Key Point Extraction

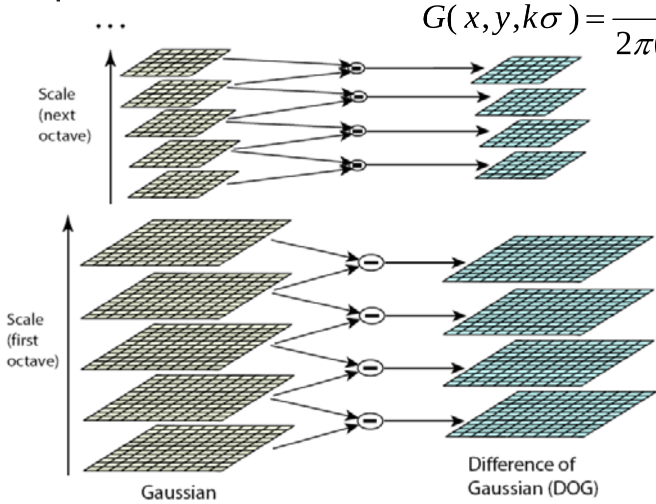


Recap
Steps for Extracting Key Points

- Scale space peak selection
  - Potential locations for finding features
- Key point localization
  - Accurately locating the feature key
- Orientation Assignment
  - Assigning orientation to the keys
- Key point descriptor
  - Describing the key as a high dimensional vector

Recap
Building a Scale Space

$$G(x,y,k\sigma) = \frac{1}{2\pi(k\sigma)^2} e^{-(x^2+y^2)/2k^2\sigma^2}$$



k is the number of chosen scale in each octave

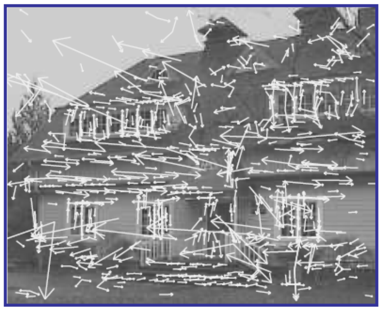
Recap
Key Point Localization

- Candidates are chosen from extrema detection

original image

extrema locations

Recap
Initial Outlier Rejection



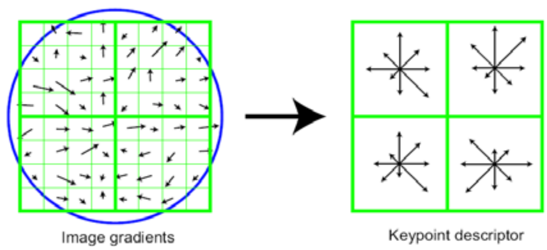

from 832 key points to 729 key points, th=0.03.

Recap
Further Outlier Rejection


from 729 key points to 536 key points.

Recap
Extraction of Local Image Descriptors at Key Points

- Compute relative orientation and magnitude in a 16x16 neighborhood at key point
- Form weighted histogram for 4x4 regions
  - Weight by magnitude and spatial Gaussian
  - Example for 8x8 to 2x2 descriptors



Recap
Extraction of Local Image Descriptors at Key Points

- Store numbers in a vector
- Normalize to unit vector (**UN**)
  - Illumination invariance (affine changes)
- For non-linear intensity transforms
  - Bound **UN** items to maximum .2
  - Renormalize to unit vector

1st Programming Project Assignment

- Implementation of SIFT key point detector
- Due date 27 February, 2006
- Deliverables:
  - Source code on a CD,
  - Project report discussing problems encountered, where the algorithm fails,
  - Images of intermediate steps
- Classroom demo is required on an unknown set of images
- Program should display computed features
- A comparison with the author's program (only executable) should show if your algorithm fails
- Look at the IJCV paper by D. Lowe

Object Detection Background Subtraction

Overview

- Scene appearance modeling
- Deviations signifies "moving" object
- Masking of moving regions
- Application of connected components

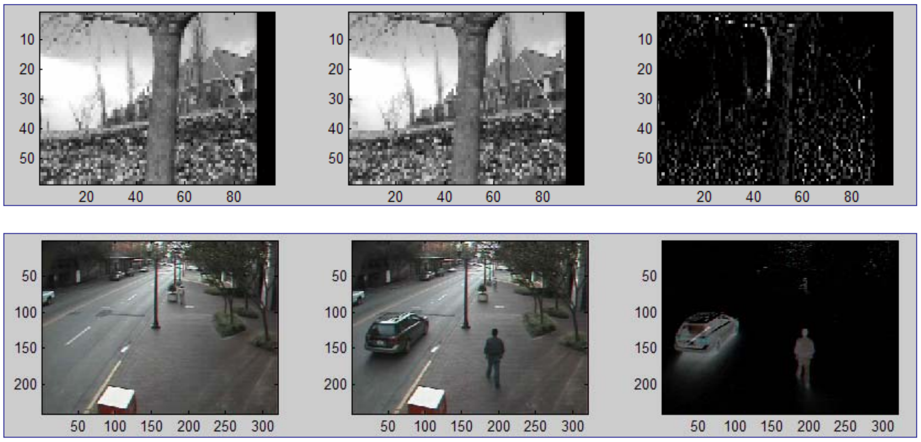
- Has been considered since 70's:
  - Jain, R. and Nagel, H. 1979. On the analysis of accumulative difference pictures from image sequences of real world scenes. PAMI 1, 2, 206–214.

Simplest Approach

- Frame differencing
  - Current frame from previous frame
    - Small average motion
  - Current frame from sample background

```
for t = 1:N
        Update_background_model()
        Compute_frame_difference()
        Threshold_frame_difference()
        Noise_removal()
end
```

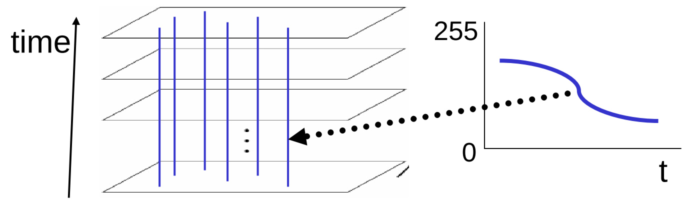
Examples



Limitations of Simple Approach

Observations

- Video is a spatio-temporal volume
  - Each pixel creates straight trajectory
  - We can look at how each ray behaves
  - What are interesting things to ask?



Observations

- Ideally: intensity is constant

- Practically it changes
  - Repetitive motion, e.g. tree motion

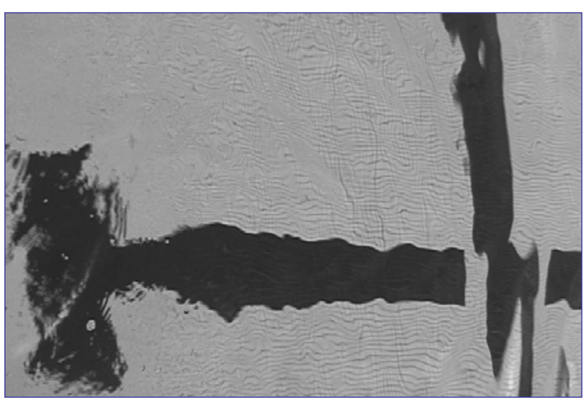
Possible Extensions

- Mean intensity, color
- Median intensity, color

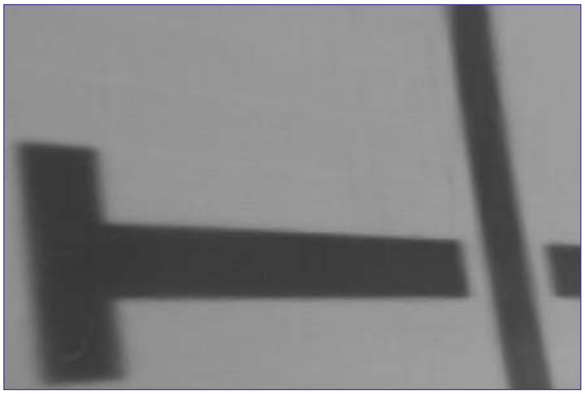
Mean color

Median color

Input Video



Average Image



What is happening?

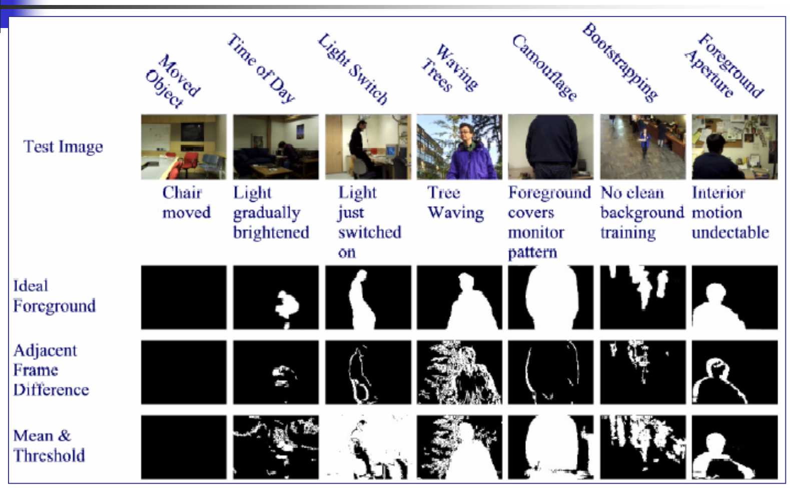
Mean Background

Subtraction

-

=

Problems With Simple Approaches



from K. Toyama et al.

Issues In General

- Noise models
  - *Unimodal:* slowly varies
  - *Multimodal:* multiple observations (moving leaf)
- Gross illumination changes
  - *Continuous:* Gradual illumination changes alter the appearance of the background
  - Discontinuous: Sudden changes in illumination alter the appearance of the background
- Bootstrapping
  - Is a training phase with “no foreground” necessary, or can the system learn what’s static vs. dynamic online?

Modeling Background

- Offline average
  - Pixel-wise mean values are computed during training phase

$$I_0(\mathbf{x}, t) = \frac{1}{T}\sum_{t=1}^{T} I(\mathbf{x}, t)$$

- Moving average
  - Background model is linear weighted sum of previous frames

$$I_0(\mathbf{x}, t) = \frac{w_a I(\mathbf{x}, t) + \sum_{i=1}^{N} w_i I(\mathbf{x}, t-i)}{w_c}$$

- Multi-Modal
  - Statistical models: Gaussian, MOG, KDE, …

$$p(I_0(\mathbf{x}, t)) = \sum_{i=1}^{n_c} \pi_i N(\mathbf{x}; \mathbf{m}_\mathbf{x}, \sigma_\mathbf{x}^2)$$

Multi Model Representation

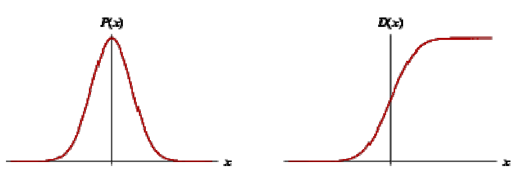
- Single Gaussian
- Mixture of Gaussian
- Kernel Density Estimates

Gaussian Distribution

Karl Friedrich Gauss
1777-1855

- The most used distribution in all of science
- Continuous distribution (unlike binomial or poisson)
- Probability *(P)* of y being in the range *[a, b]* is given by an integral

$$P(a < y < b) = \int_a^b p(y)dy = \frac{1}{\sigma\sqrt{2\pi}} \int_a^b e^{-\frac{(y-\mu)^2}{2\sigma^2}} dy$$



68% of area is within $\pm\sigma$

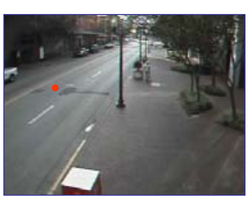
Why is Gaussian Everywhere?

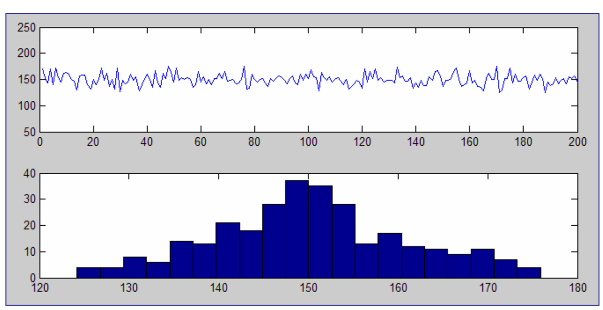
- **Central Limit Theorem:**
  - Crude statement: Things that are the result of the addition of lots of small effects tend to become Gaussian.
- Let *Y*1, *Y*2,...*Y*n be an infinite sequence of independent random variables each with the same probability distribution.
  - Suppose that the mean (μ) and variance (σ2) of this distribution are both finite. For any numbers *a* and *b*:

$$\lim_{n\to\infty} P\left[a < \frac{Y_1 + Y_2 + ...Y_n - n\mu}{\sigma\sqrt{n}} < b \right] = \frac{1}{\sqrt{2\pi}} \int_a^b e^{-\frac{1}{2}y^2}\, dy$$

In Context Of Background Subtraction

- Intensity variation is around some intensity





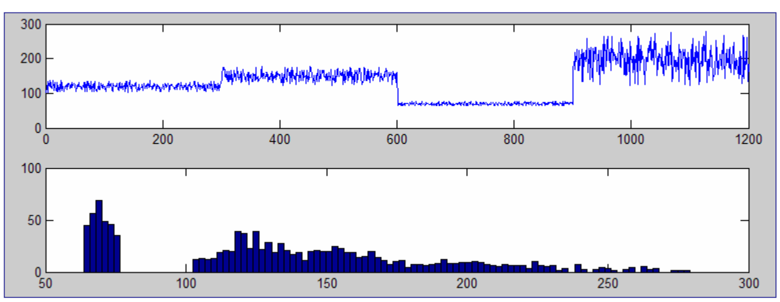
Wren, C., Azarbayejani, A., and Pentland, A. 1997. Pfinder: Real-time tracking of the human body. PAMI 19, 7, 780–785.

Estimation of Model Parameters N(μ,σ)

- Compute mean
- Compute standard deviation
- Simple to implement

Mixture Of Gaussians

- Single Gaussian: assumption is single cluster
- Data may consist of multiple clusters



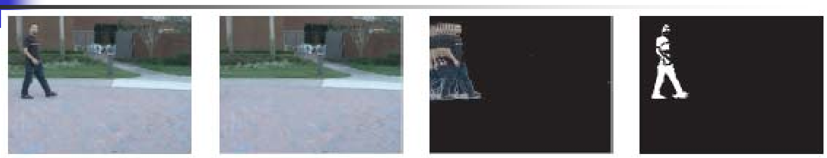
Stauffer, C. and Grimson, W. 2000. Learning patterns of activity using real time tracking. PAMI 22, 8, 747–767.

Application To Background Modeling

- Select number of mixtures components N
- Estimation
  - Expectation maximization: not real time!
  - Crude approximation
    - Initialize 1st component with input image, others with 0 mean, constant covariance
    - Comparing pixel with every Gaussian until a match is found
    - If a match is found the mean and variance is updated,
    - Otherwise start a new Gaussian with mean equal to the current pixel color and some initial variance

A.P. Dempster, N.M. Laird, and D.B. Rubin (1977): "Maximum Likelihood from Incomplete Data via theEM algorithm", *Journal of the Royal Statistical Society*, Series B, vol. 39, 1:1-38

Example



(a) (b) (c) (d)

- (a) Image from a sequence
- (b) The mean of the highest weighted Gaussians at each pixels position.
- (c) The means of the Gaussian with the second highest weight, these means represent colors that are observed less frequently.
- (d) Background subtraction result.
  - The foreground consists of the pixels in the current frame that matched a low weighted Gaussian

Possible Student Presentations

- Graph-Cut
  - Shi & Malik, "*Normalized Cuts and Image Segmentation*" TPAMI, 1997
- Adaboost
  - Viola & Jones, "Rapid Object Detection using a Boosted Cascade of Simple Features", CVPR 2001
- Support Vector Machines
  - Papageorgiou et al., "A General Framework for Object Detection", ICCV 1998