Background Difference

Lecture-10

Motivation

- Detection of interesting objects in videos is the first step in the process of automated surveillance.
- Focus of attention method greatly reduces the processing time required for tracking and activity recognition.

Introduction

- Objectives:
  - Given a sequence of images from a stationary camera identify pixels comprising 'interesting' objects.
  - All independently moving objects are interesting!
- General Solution
  - Model properties of the scene (e.g. color, texture e.t.c) at each pixel.
  - Significant change in the properties indicates an interesting change.

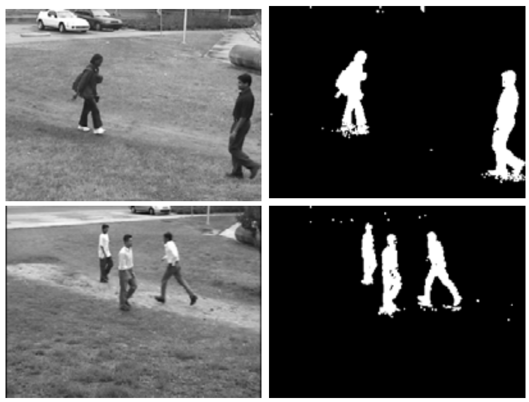
UCF Computer Vision Lab. 3

Introduction

- Problems in Realistic situations:
  - Moving but uninteresting objects
    - e.g. trees, flags or grass.
  - Long term illumination changes
    - e.g. time of day.
  - Quick illumination changes
    - e.g. cloudy weather
  - Shadows
  - Other Physical changes in the background
    - Dropping or picking up of objects
  - Initialization

UCF Computer Vision Lab. 4

Segmenting Background

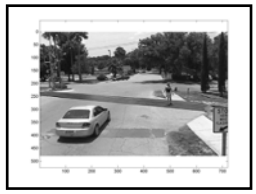


Difference Pictures

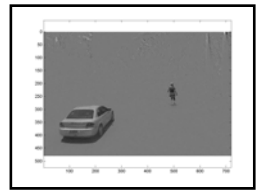
- Jain, R. and Nagel, H. 1979. ``On the analysis of accumulative difference pictures from image sequences of real world scenes”. *IEEE Trans. on Pattern Analysis and Machine Intelligence 1,* 2, pp 206-214.

Background Subtraction

-



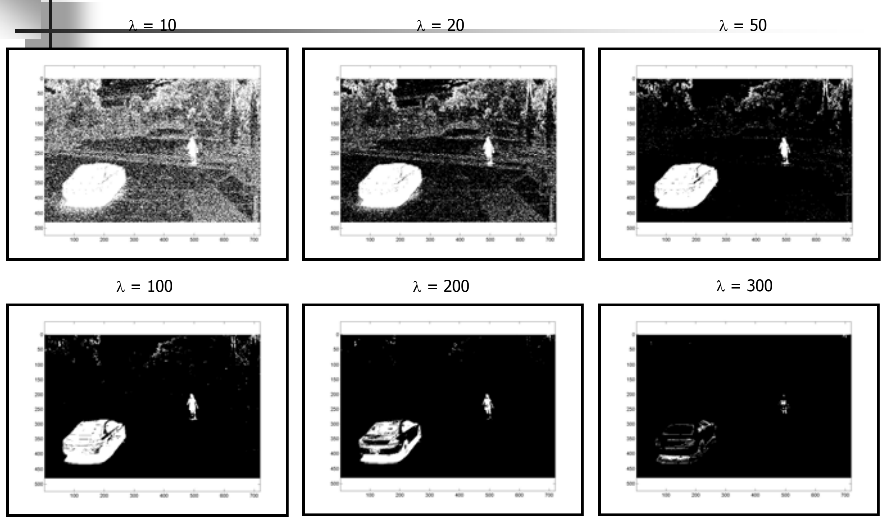
=



- Problem: Choosing a threshold
  - Pixel is foreground if $I_1(x,y)-I_2(x,y) \leq \lambda$ otherwise background?
  - What is the correct value of λ ?

UCF Computer Vision Lab. 7

Setting a Threshold



UCF Computer Vision Lab. 8

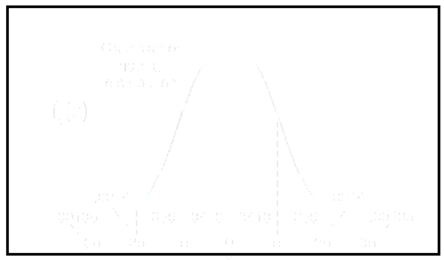
MODELING PIXEL INTENSITIES WITH A NORMAL DISTRIBUTION

Each pixel intensity can be modeled by a Normal Distribution, defined in terms of a mean μ and variance σ^2, as $N(\mu, \sigma^2)$. μ and σ are called parameters.

Useful when you wish to establish membership of a pixel to one of several models.

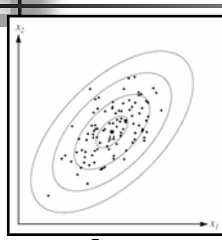
$N(\mu, \sigma^2)$ is a probability distribution function defined by:

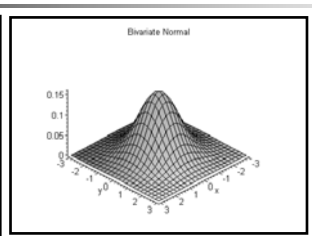
$$f(\mathbf{x} \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-(x-\mu)^2}{(2\sigma^2)}}$$

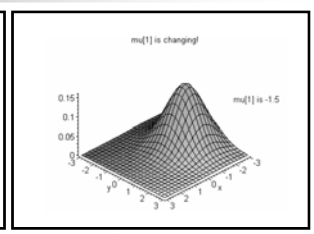


UCF Computer Vision Lab. 9

Bi-variate Normal Distribution







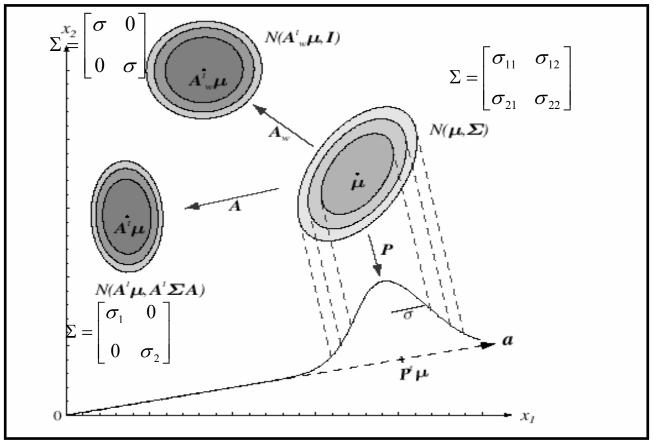
- If we were interested in *r-g*, or *g-b*, or *r-b*…
- The mean can be updated over time simply as

$$\mu_t = (1-\rho)\mu_{t-1} + \rho X_t$$

$$\sigma_t^2 = (1-\rho)\sigma_{t-1}^2 + \rho(X_t - \mu_t)^T (X_t - \mu_t)$$

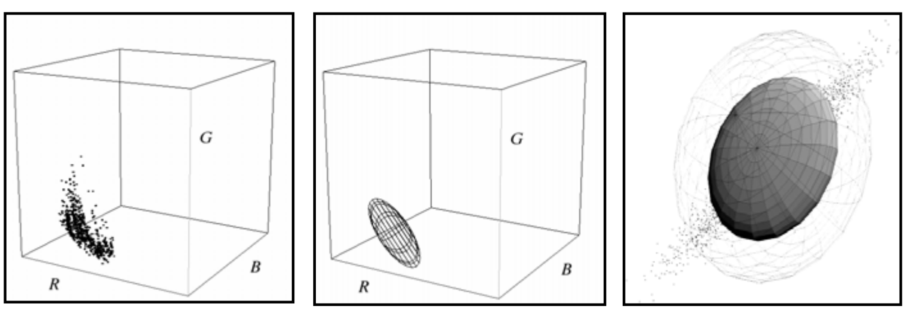
UCF Computer Vision Lab. 10

Covariance



UCF Computer Vision Lab. 11

Tri-variate Normal



UCF Computer Vision Lab. 12

Method I: *P*finder

- *P*finder (Person Finder) - Wren *et al* of MIT (1997)
- C. Wren, A. Azarbayejani, T. Darrel, and A. Pentland, "*Pfinder: Real time Tracking of the Human Body*," IEEE Transactions on Pattern Analysis and Machine Intelligence, 1997.
- Color of each pixel modeled as a three-dimensional Gaussian.
- Big Advantage: adaptivity, pixel-wise 'threshold'.

The Bottom Line

- Model each pixel color as a three dimensional normal distribution
- Adapt the color means and variances over time
- Slowly changing illuminations are handled
- Changes to background are eventually learnt
- Relocation and initialization problems are eventually learnt too.

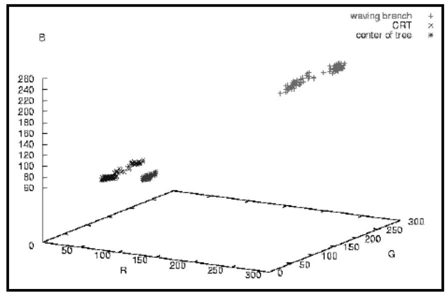
Limitations

- Unfortunately, this method has limitations
- Due to dynamic nature of real-world scenes modeling pixels with single Gaussian distributions is inaccurate
- Quick illumination changes are not handled
- Good for indoor scenes

Method II: Mixture of Gaussians

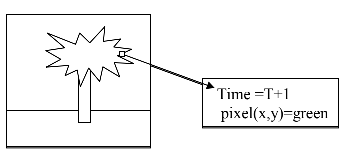
- *MoG* (Mixture of Gaussians) - Stauffer *Grimson* of MIT (2000)
- C. Stauffer, E. Grimson, "*Learning Patterns of Activity using Real-time Tracking*," IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000
- Color of each pixel modeled as a *mixture* of three-dimensional Gaussian
- Big Advantage: Handles Multimodality

Multimodality



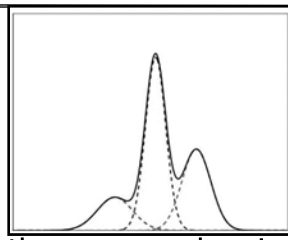
A Background Subtraction Method by Stauffer and Grimson

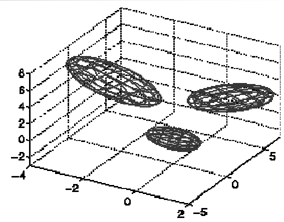
- In realistic scenarios multiple Processes are generating color '*x*' at each pixel, where x=[R,G,B]^T



- A method is required that can incorporate multiple colors in the background model.

Mixture of Gaussians





- Finding the mean and variance for one Gaussian is easy
- Much tougher for the Mixture of Gaussians case
- Find
  - Number of Gaussians
  - Mean and variance of each Gaussian

UCF Computer Vision Lab. 19

A Background Subtraction Method by Stauffer and Grimson

- For each pixel *(i,j)* at time '*t*' each process is modeled as a Gaussian distribution.
  - Gaussian distribution is described by a mean 'm' and a covariance matrix Σ.

$$N(x_{i,j}^t \mid m_{i,j}^t, \Sigma_{i,j}^t) = \frac{1}{(2\pi)^{\frac{n}{2}} \mid \Sigma_{i,j}^t \mid} e^{-\frac{1}{2}(x_{i,j}^t - m_{i,j}^t)^T (\Sigma_{i,j}^t)^{-1} (x_{i,j}^t - m_{i,j}^t)}$$

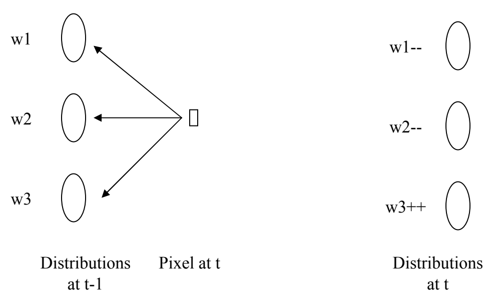
  - Weight ω associated with each distribution signifying relevance in recent time.

- Thus each Pixel is modeled as a mixture of Gaussians.

UCF Computer Vision Lab. 20

A Background Subtraction Method by Stauffer and Grimson

- At each frame
  - Calculate mahalanobis distance of pixel's color value from each of the associated K Guassian distributions



A Background Subtraction Method by Stauffer and Grimson

– If a match is found with the k^{th} Gaussian, update parameters

$$m_{i,j}^{t,k} = (1-\rho)m_{i,j}^{t-1,k} + \rho x_{i,j}^{t}$$

$$\Sigma_{i,j}^{t,k} = (1-\rho)\Sigma_{i,j}^{t-1,k} + \rho(x_{i,j}^{t} - m_{i,j}^{t})(x_{i,j}^{t} - m_{i,j}^{t})^{T}$$

•where *p* is a learning parameter

A Background Subtraction Method by Stauffer and Grimson

– If a match is not found
  – Replace lowest weight distribution with a new distribution such that

$$m_{i,j}^{t,new} = x_{i,j}^{t}$$

$$\sum_{i,j}^{t,new} = \sum^{initial}$$

- The prior weights of K distributions are adjusted as

$$\omega_{i,j}^{t-1} = (1-\alpha)\omega_{i,j}^{t-1} + \alpha(M_{i,j}^{t-1})$$

  • M is1 for model that matched and 0 for others

- Foreground= Matched distributions with weight< T` + Unmatched pixels

UCF Computer Vision Lab. 23

Background Modeling using Mixture of Gaussians

- Pros
  - Handles slow changes in illumination conditions
  – Can accommodate physical changes in the background after a certain time interval.
  –Initialization with moving objects will correct itself after a certain time interval.

UCF Computer Vision Lab. 24

Background Modeling using Mixture of Gaussians

- Cons
  - Can't handle quick changes in illumination conditions e.g. cloudy weather.
  - Initialization with moving objects
  - Physical changes in background
  - Shadows

UCF Computer Vision Lab. 25

Results

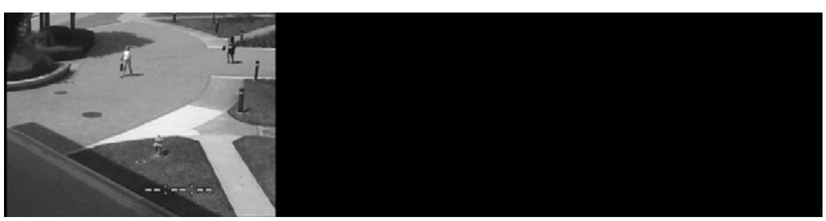
UCF Computer Vision Lab. 26

Results

Stauffer & Grimson

Javed, Khurram & Shah

Results



Stauffer & Grimson

Javed, Khurram & Shah

Summary of Algorithm

- Learn background model by watching 30 second video
- Detect moving object by measuring deviations from background model, and applying connected component to foreground pixels.
- Update background and blob statistics

UCF Computer Vision Lab. 29

Kanade

Summary

- Very similar to k-Gaussian with following differences:
  - uses only single Gaussian
  - uses gray level images, the mean and variance are scalar values

UCF Computer Vision Lab. 31

Algorithm

- Learn background model by watching 30 second video
- Detect moving object by measuring deviations from background model, and applying connected component to foreground pixels.
- Update background and region statistics

UCF Computer Vision Lab. 32

Detection

- During detection if intensity value is more than two sigma away from the background it is considered foreground:
  - keep original mean and variance
  - track the object with new mean and variance
  - if new mean and variance persists for sometime, then substitute the new mean and variance as the background model
  - If object is no longer visible, it is incorporated as part of background

W4 (Who, When, Where, What)

Davis

W4

• Compute "minimum"(M(x)), "maximum" (N(x)), and "largest absolute difference" (L(x)).

$$D_i(x,y) = \begin{cases} 1 \quad if \quad |M(x,y) - f_i(x,y)| > L(x,y) or \\ \qquad |N(x,y) - f_i(x,y)| > L(x,y) \\ \qquad 0 \quad \ldots \quad otherwise \end{cases}$$

- Theoretically, the performance of this tracker should be worse than others.
- Even if one value is far away from the mean, then that value will result in an abnormally high value of L.
- Having short training time is better for this tracker.

Limitations

- Occlusion
- Shadows
- Slow moving people
- Multiple processes (swaying of trees..)
- Quick Illumination Changes

Webpage

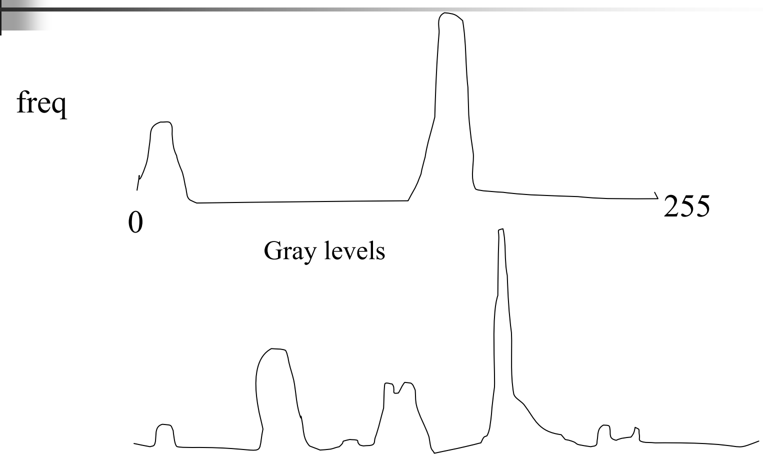
- Http://www.cs.cmu.edu/~vsam

Skin Detection

Kjeldsen and Kender

Training

- Crop skin regions in the training images.
- Build histogram of training images.
- Ideally this histogram should be bi-modal, one peak corresponding to the skin pixels, other to the non-skin pixels.
- Practically there may be several peaks corresponding to skin, and non-skin pixels.



Training

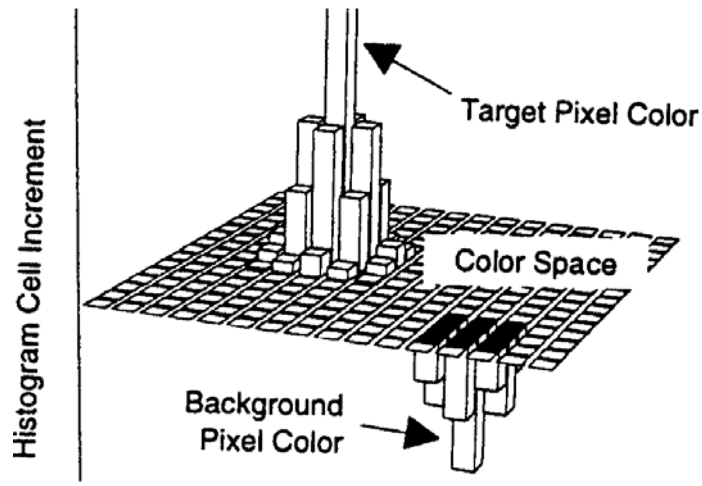
- Apply threshold to skin peaks to remove small peaks.
- Label all gray levels (colors) under skin peaks as "skin", and the remaining gray levels as "non-skin".
- Generate a look-up table for all possible colors in the image, and assign "skin" or "non-skin" label.

Detection

- For each pixel in the image, determine its label from the “look-up table” generated during training.

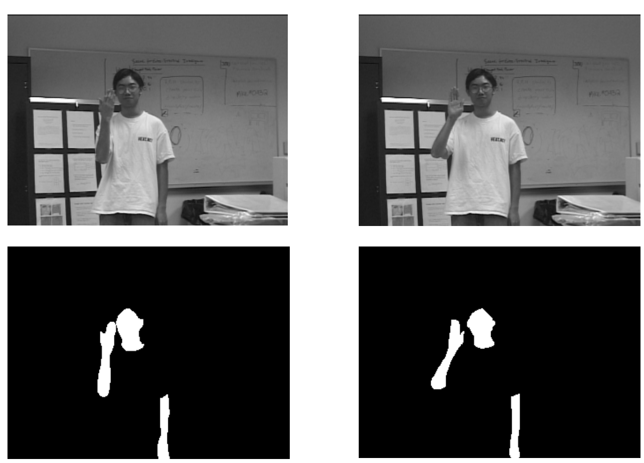
Building Histogram

- Instead of incrementing the pixel counts in a particular histogram bin:
  - for skin pixel increment the bins centered around the given value by a Gaussian function.
  - For non-skin pixels decrement the bins centered around the given value by a smaller Gaussian function.



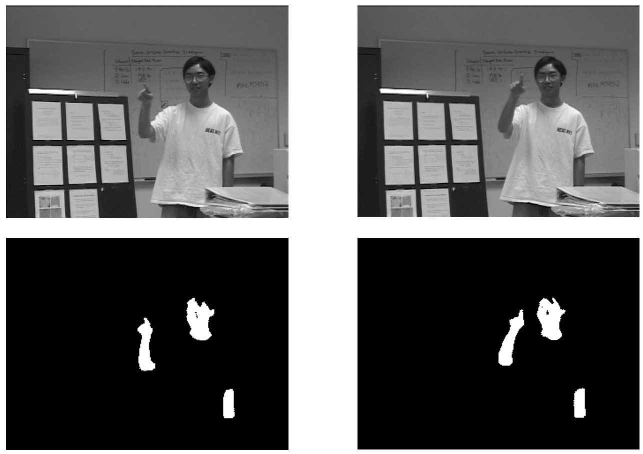
45

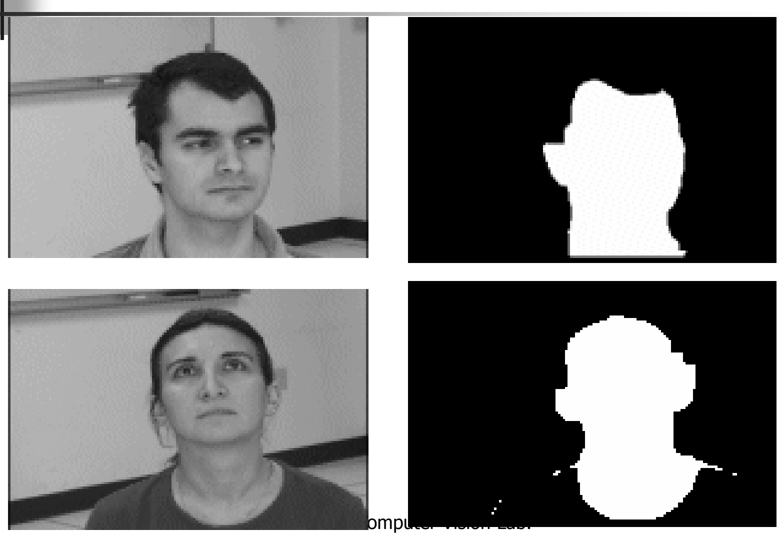
Example training images



UCF Computer Vision Lab. 46

Results of skin detection





Results of skin detection

Detecting Fire