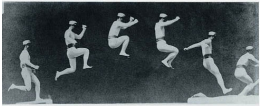
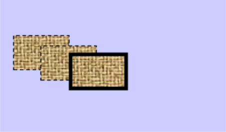
Sorbonne Université
M2 IMA - "UE VISION"
Motion detection in videos

Antoine Manzanera
ENSTA-Paris



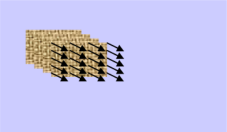
Motion Detection and Video Analysis

Three kinds of image processing primitives in Video analysis:



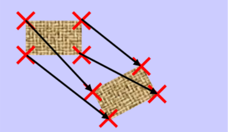
Detection

Separate mobile pixels from the static background



Estimation

Calculate the apparent velocity of each pixel



Tracking

Match spatial structures from frame to frame

Content and Goals of the lecture

- Present the characteristics, challenges and difficulties of mobile objects detection in image sequences.
- Explain the different techniques of background modelling used in temporal change detection.
- Briefly expose some spatiotemporal regularisation methods related to motion detection.

Lecture outline

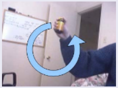
Application fields

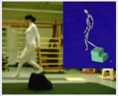
Smart videosurveillance

- Geofencing / Abnormal activity
- Aggression / distress detection / crowd surveillance
- Dynamic (e.g. gait) biometry

Human-Machine Interfaces

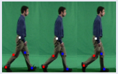
- Visual command
- Avatar control
- Language sign

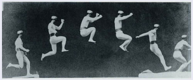




Bio-medical applications

- Gait analysis
- Elderly monitoring
- Sport analysis





Motion segmentation

Context

- Stationary camera
- Uncontrolled acquisition

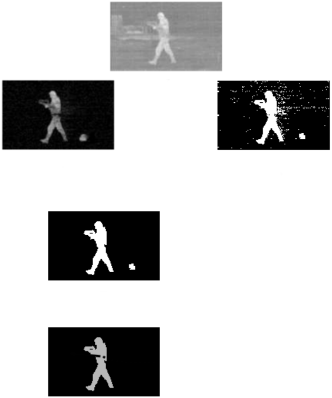
Background segmentation

Objective: Separate the moving object (foreground) from the static scene (background).

- Robust estimation problem
- Temporal statistics representation
- Computational cost: Space and Time complexities

Detection: global view

1. **Temporal change estimation:** Temporal statistics are calculated on every pixel, from which outlier values can be deduced.
2. **Spatiotemporal regularisation:** The results are aggregated to form regular shapes.
3. **Objects selection:** The obtained regions are selected according to morphological or kinematic criteria.



Which observations?

What kind of temporal variation shall we consider?

Temporal gradient

$D_t = |I_t - I_{t-1}|.$

- ⊕ Very simple!
- ⊕ Very adaptive!
- ⊖ Aperture problem!

Marginal values

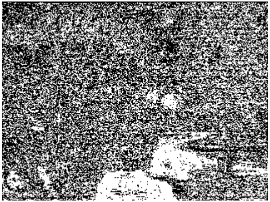
$D_t = |I_t - B_t|.$

- ⊕ Aperture problem
- ⊕ Complex background management
- ⊖ Adaptation is trickier

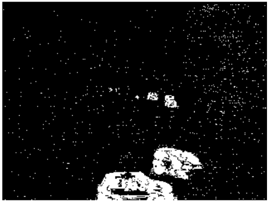
Temporal gradient

I_t (256 gray levels)

$D_t = |I_t - I_{t-1}|$



Threshold D_t to 3



Threshold D_t to 9

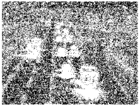
Setting the threshold

The *global* level of threshold may be *dynamically* adjusted by:

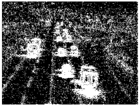
1. Assuming that isolated points are only due to noise.
2. Setting a target rate r_{target} of isolated points.

Let r the rate of isolated points in the binary image.

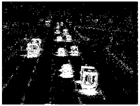
If $r < r_{target}$ then $\tau_t \leftarrow \tau_{t-1} - 1$, else $\tau_t \leftarrow \tau_{t-1} + 1$.



$\tau_t = 2$

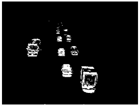


$\tau_t = 4$



$\tau_t = 8$

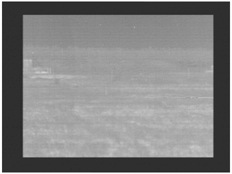
$\tau_t = 15$



$\tau_t = 25$

Static background estimation

Video

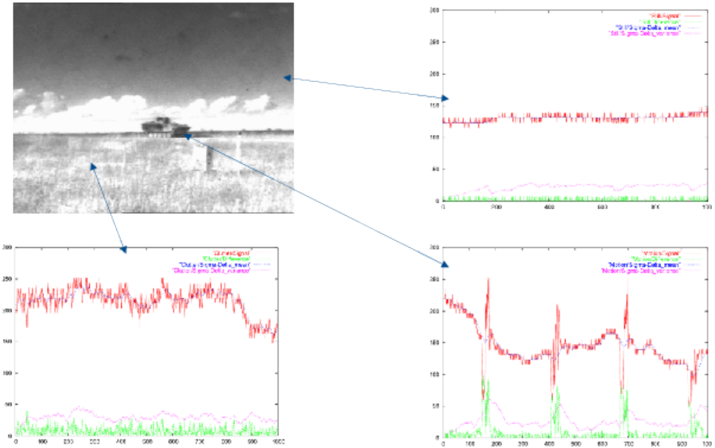


Background (static)

Foreground (mobile objects)

- Temporal series processing
- Non stationary estimation
- Foreground/Background classification

A robust estimation problem...



Temporal average?

Naive recursive average

$$B_t = \frac{1}{t} I_t + \frac{t-1}{t} B_{t-1}$$

- Recursive computation of the arithmetic average
- Not computable for large values of t!

Temporal average

Exponential filter

$$B_t = \alpha I_t + (1-\alpha)B_{t-1} \; ; \; \alpha \in]0,1[$$

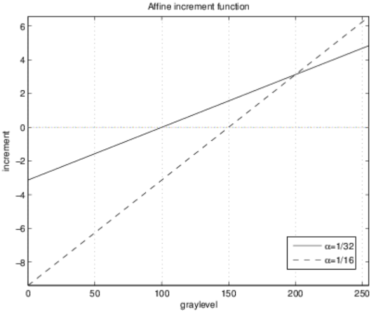
- α is the learning rate ; $\alpha \approx \frac{1}{t}$
- If $\alpha = 2^{-N}$: very efficient computation
- Incremental formulation: $B_t = B_{t-1} + \alpha(I_t - B_{t-1})$

General incremental formulation

Recursive estimation of the background (1st order)

$$B_t = B_{t-1} + \delta_t(I_t, B_{t-1})$$

For the exponential filter:
$\delta_t(I_t, B_{t-1}) = \alpha(I_t - B_{t-1})$
The increment function is linear...
Figure: 2 examples of increment functions for the exponential filter.



Bi-level exponential filter

Bi-level temporal average

$B_t = B_{t-1} + \alpha_1 (I_t - B_{t-1})$; if $I_t \in$ Background
$B_t = B_{t-1} + \alpha_2 (I_t - B_{t-1})$; if $I_t \in$ Foreground ($\alpha_2 << \alpha_1$)

A classification criterion is then necessary.
E.g., a threshold:
$|I_t - B_{t-1}| > \tau_t$

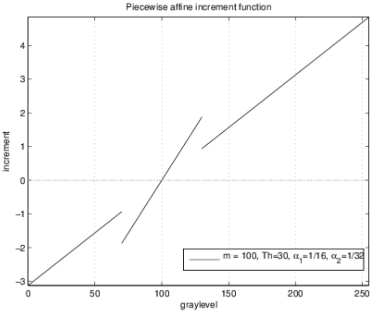


Figure: 1 example of increment function for the bi-level exponential filter.

Recursive estimation of the average and variance

The same recursive scheme can be used to estimate the temporal variance, which allows to *locally adjust* the classification Foreground/Background threshold:

Recursive Average and Variance

$D_t = I_t - B_{t-1}$
If $|D_t| > n\sqrt{V_{t-1}}$, $E_t = 1$ (Foreground), else $E_t = 0$ (Background).
$B_t = B_{t-1} + \alpha_t D_t$
$V_t = V_{t-1} + \alpha_t D_t^2$

- B_t is the average, V_t the variance.
- n is an integer, typically 2 or 3.
- $\alpha_t = \alpha_1$ if $E_t = 0$, and $\alpha_t = \alpha_2$ otherwise ($\alpha_2 << \alpha_1$).

Recursive estimation of the average and variance

Recursive Average and Variance

$D_t = I_t - B_{t-1}$

If $|D_t| > n\sqrt{V_{t-1}}$, $E_t = 1$ (Foreground), else $E_t = 0$ (Background).

$B_t = B_{t-1} + \alpha_t D_t$

$V_t = V_{t-1} + \alpha_t D_t^2$

Estimating the variance allows to locally adapt the threshold, however the increment function remains linear (α) and/or discontinuous ($\alpha_2 < \alpha_1$).

Estimation weighted by the density

In fact, considering the incremental expression $B_t = B_{t-1} + \delta_t(I_t, B_{t-1})$, the increment function δ_t should also depend on the probability to observe the value I_t:

Weighted estimation (general case)

$$\delta_t(I_t, B_{t-1}) = \frac{\alpha_{max} f_t(I_t)}{f_t(B_{t-1})} \times (I_t - B_{t-1})$$

with:

- $f_t(x) = P(B_t = x)$ probability density of the background.
- α_{max} maximal learning rate.
- B_{t-1} corresponds to the current mode of the distribution.

Temporal density estimation

The temporal density can be estimated using the recursive histogram update method:

Temporal density estimation

- Let $\{1, \dots, N\}$ be the histogram bins.
- Initialization: $f_0(i) = 1/N$ for every $i \in \{1, \dots, N\}$
- For $t > 0$:
  - $f_t(I_t) = f_{t-1}(I_t) + \varepsilon$
  - Renormalize f_t

The reference value of the background B_t can (if necessary) be defined as the *mode* of the histogram $\arg\max_{i \in \{1,\dots,N\}} f_t(i)$, or as the *median* value, using $F_t^{-1}(1/2)$, where $F_t(i) = \sum_{j<i} f_t(i)$.

Temporal density estimation

Temporal density estimation

- Let $\{1, \ldots, N\}$ be the histogram bins.
- Initialization: $f_0(i) = 1/N$ for every $i \in \{1, \ldots, N\}$
- For $t > 0$:
  - $f_t(I_t) = f_{t-1}(I_t) + \varepsilon$
  - Renormalize f_t

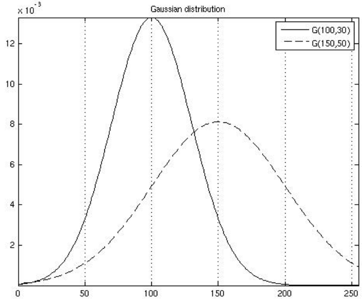
The classification can also be made directly (i.e. without estimating the reference background B_t), from the density, for example: if $f_t(I_t) < \tau$, then $E_t = 1$.

Estimation of Gaussian density

If the density corresponds to a known model, the estimation can be simplified, for example in the case of a single Gaussian (1 mode/average, 1 variance) :

Gaussian distribution

$$f_t(x) = \frac{1}{\sigma_t\sqrt{2\pi}} \exp\left(-\frac{(x-\mu_t)^2}{2\sigma_t^2}\right)$$



Estimation of Gaussian density

Gaussian increment function

$$\delta_t(I_t, B_{t-1}) = \alpha_{max} \times \exp\left(\frac{-(I_t - B_{t-1})^2}{2V_{t-1}}\right) \times (I_t - B_{t-1})$$

Variance estimation:
$V_t = V_{t-1} + \alpha_V((I_t - B_t)^2 - V_{t-1})$
Classification:
$E_t = 1 \Leftrightarrow |I_t - B_t| > k \times \sqrt{V_t}$

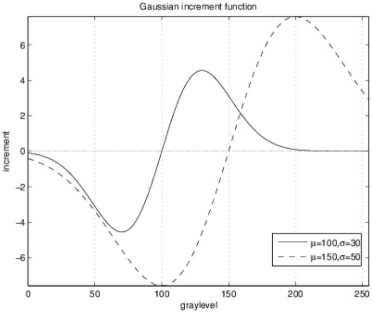


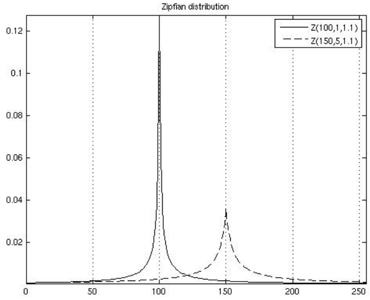
Figure: 2 examples of increment functions for a Gaussian density.

The Zipf-Mandelbrot distribution

Centred Zipfian Distribution

$$Z_{(\mu,k,s)}(x) = \frac{(s-1)k^{s-1}}{2(|x-\mu|+k)^s}$$

- μ is the average (mode) of the distribution
- k determines the dispersion ($\simeq$ variance)
- $s \simeq 1; s > 1$



The Zipf-Mandelbrot distribution

Centred Zipfian Distribution

$$Z_{(\mu,k,s)}(x) = \frac{(s-1)k^{s-1}}{2(|x-\mu|+k)^s}$$

- Origin: linguistics (frequence of words in most languages).
- Has been used in spatial image processing (coding, segmentation).
- Used here as a temporal distribution model.

Zipfian background estimation

Zipfian increment function

The Zipfian increment function can be approximated by a Heaviside function:

$$\delta_t \simeq H_{(\mu,\kappa)}(x) = -\kappa \text{ if } x < \mu,\ +\kappa \text{ if } x > \mu \text{ (with } \kappa = \alpha_{max} k^s)$$

Thus, the Zipfian estimation can be approximated by the Σ-Δ modulation:

$$B_t = B_{t-1} + \varepsilon \text{ if } I_t > B_{t-1}$$
$$B_t = B_{t-1} - \varepsilon \text{ if } I_t < B_{t-1}$$

But the elementary increment ε should depend on the variance of the background.

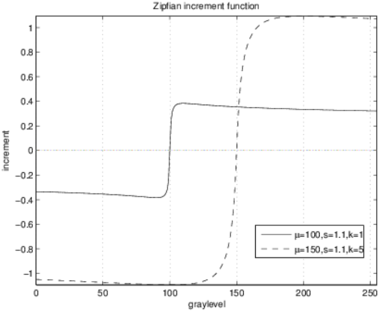
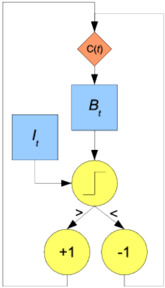


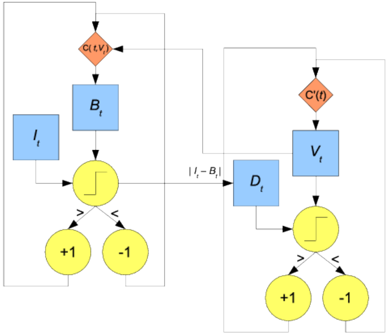
Figure: 2 examples of increment functions for a Zipfian density.

Σ-Δ estimation algorithm (1)



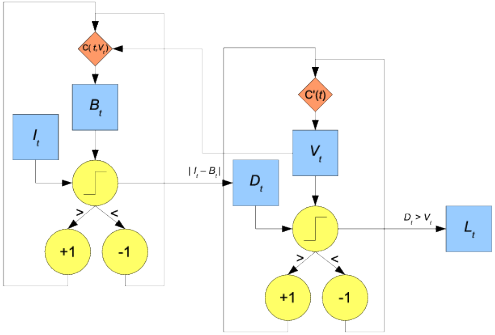
The elementary increment corresponds to the Least Significant Bit (LSB), i.e. ± 1.
The average increment is temporally adjusted by changing the update frequence:
This corresponds to the condition $C(t)$ (typically $C(t) \equiv (t\%n) == 0$)

Σ-Δ estimation algorithm (2)



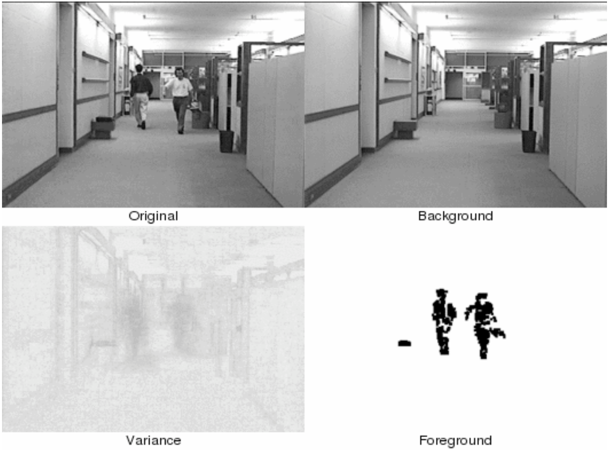
As the average increment should depend on the variance of the background, the update condition should also depend on the dispersion estimator V_t. (The larger V_t, the more frequent the update).
The dispersion estimator V_t is also calculated by Σ-Δ estimation, based on the absolute difference sequences $|I_t - B_t|$.

Σ-Δ estimation algorithm (3)



Finally, the classification Foreground/Background is simply obtained by comparing the absolute difference to the current dispersion estimate.

Example: Sequence with radial motion



Quantitative evaluation

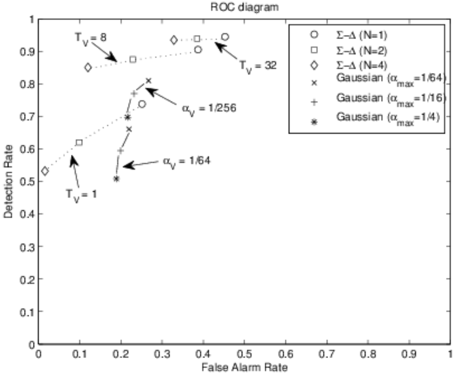


Figure: Comparison of several background subtraction algorithms based on Σ-Δ or Gaussian estimation, using different temporal parameters.

Computational advantages

- The computational cost of Σ-Δ is extremely low:
  - *Memory*: 2 integers per pixel.
  - *Instruction set*: reduced to difference, comparison, and increment/decrement.
  - *Data size*: No approximation, adapted to Fixed-Point Arithmetic of any size.
- It was implemented on various embedded platforms, like:
  - *Cellular parallelism*: Programmable retina *PVLSAR 34*.
  - *Vector parallelism*: Multimedia extensions *SSE2, Altivec*.
  - *Programmable Components*: FPGA *Xilinx XSA3S1000*.

Multi-modal background estimation

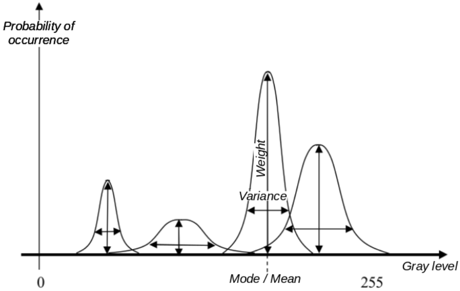
The use of mono-modal distributions as probabilistic model can be irrelevant in the case of complex background (e.g. sea waves, moving flags,...). However, the previous methods can be extended to multi-modal (mixture) models, as follows:

Multi-modal background estimation

Let $\{B^i, V^i, W^i\}_{i=1..N}$ represent the N modes
For every pixel I_t, for every mode i:
if $|I_t - B_t^i| < n\sqrt{V_t^i}$:
Update the corresponding $\{B_t^i, V_t^i, W_t^i\}$ (B^i, V^i updated as in the monomodal case, W_t^i is incremented then normalized)
Rank the different modes according to their "importance" $W^i/\sqrt{V^i}$, and choose the first ones as background.

Multi-modal background estimation

- The multi-modal distribution is represented by $3N$ scalar values $\{B^i, V^i, W^i\}_{i=1..N}$ per pixel.
- N the number of modes, is typically between 3 and 7.
- B^i and V^i represent the average (mode) and variance of each sub-distribution.
- W^i represent the relative weights of the different modes.

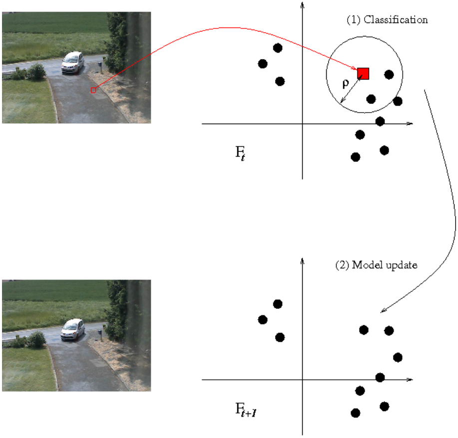


Sample-Consensus methods

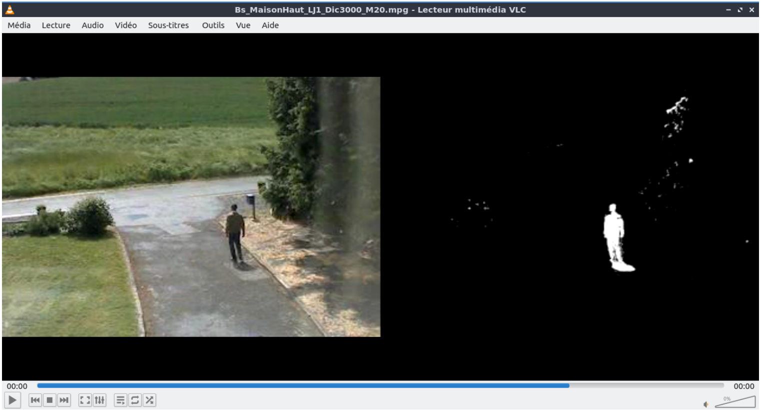
- Some methods represent the background without calculating explicitly statistics, but by keeping in memory some values $\{I_{t_1}, \ldots, I_{t_K}\}$ (*sampling*).
- Foreground/Background classification is performed by deciding whether the current value is close to the sample or not (*consensus*). Example ViBe: $E_t = 1 \Leftrightarrow |\{i \in \{1, \ldots, K\}; d(I_t, I_{t_i}) > \tau\}| > T$.
- The sample is then updated, possibly by considering the value of E_t.

Sample-Consensus methods

The Sample-Consensus methods can be applied on the gray level, on multidimensional colour spaces, or even on local feature spaces (e.g. filter banks, or deep features...).



Example: Feature-ViBe

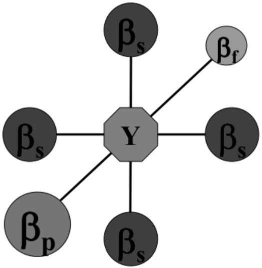


Markovian regularization

Temporal change detection is not sufficient to perform mobile object segmentation. Spatiotemporal regularization based on Markov fields has been used for mobile objects detection:

- **Modelling:** the Fixed/Mobile binary label is assumed to be a Markov field in the discrete space-time.
- **Hammersley-Clifford theorem:** the density can be calculated from a function (energy) defined on the cliques of the discrete mesh.
- **Simulation:** some samples of this random field can be obtained (e.g. Gibbs sampler).
- **Optimisation:** to find the most likely realisation of this field (e.g. ICM, Simulated annealing).

Markovian regularization: Modelling the Gibbs Energy



$$\underbrace{U(x)}_{\text{Energy}} = \underbrace{U_m(x)}_{\text{Model}} + \underbrace{U_a(x,y)}_{\text{Data}}$$

x: binary (B/F) label image (E_t).
y: absolute difference image ($|D_t|$).

Model energy term (Potts Model)

$$U_m(x) = \sum_{s \in \mathbb{S}} \sum_{r \in \mathcal{V}(s)} V_x(s,r)$$

with $V_x(s,r) = -\beta_{sr}$ if $x(s) = x(r)$, $+\beta_{sr}$ otherwise, and $\beta_{sr} > 0$.

Data energy term

$$U_a(x,y) = \frac{1}{2\sigma^2} \sum_{s \in \mathbb{S}} y(s) - \alpha x(s)$$

with $\alpha > 0$.

Markovian regularization: Modelling the Gibbs Energy

Model Energy Term

$$U_m(x) = \sum_{s \in \mathbb{S}} \sum_{r \in \mathcal{V}(s)} \pm \beta_{sr}$$

The B/F label image X is assumed to be a Markov field:

$$P(X = x) = \frac{e^{-U_m(x)}}{Z_1}$$

The Model energy expresses a regularity hypothesis.

Data Energy Term

$$U_a(x, y) = \frac{1}{2\sigma^2} \sum_{s \in \mathbb{S}} y(s) - \alpha x(s)$$

The observation (difference) image Y is assumed to be related to X by:

$$P(Y = y / X = x) = \frac{e^{-U_a(x,y)}}{Z_2}$$

Where α and σ are the mean and standard deviation of Y.

Markovian regularization: Bayesian labelling

Model Energy Term

$$U_m(x) = \sum_{s \in \mathbb{S}} \sum_{r \in \mathcal{V}(s)} \pm \beta_{sr}$$

$$P(X = x) = \frac{e^{-U_m(x)}}{Z_1}$$

Data Energy Term

$$U_a(x, y) = \frac{1}{2\sigma^2} \sum_{s \in \mathbb{S}} y(s) - \alpha x(s)$$

$$P(Y = y/X = x) = \frac{e^{-U_a(x,y)}}{Z_2}$$

Bayesian labelling: *Maximum A Posteriori* criterion

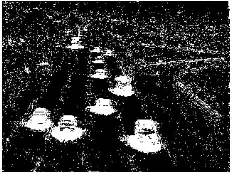
$$\begin{aligned} \arg\min_x U(x) &= \arg\max_x P(X = x) P(Y = y/X = x) \\ &= \arg\max_x P(X = x/Y = y) \end{aligned}$$

[Bouthémy93]

Regularization by Spatiotemporal Morphology

Space-time regularization is often performed on binary images of Foreground using the operators from Mathematical Morphology:

- Alternated Sequential Filters (ASF):
  $F_n(E_t) = \delta_{B_n}(\varepsilon_{B_n}(\delta_{B_{n-1}}(\varepsilon_{B_{n-1}}(\ldots\delta_{B_1}(\varepsilon_{B_1}(E_t))\ldots)))).$



E_t

$F_2(E_t)$

Regularization by Spatiotemporal Morphology

Connected Morphological operatosrs:

- ASF by reconstruction: $E'_t = R_{E_t}(F_n(E_t))$.

E_t

$F_2(E_t)$

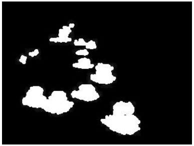
E'_t

Regularisation by Spatiotemporal Morphology

Spatiotemporal connected operators:

- Spatiotemporal connected filter:
  $E_t'' = R_{E_t}\left(F_n(E_t) \cap \delta_{B_m}(E_{t-1}')\right).$

E_{t-1}'



$\delta_{B_m}(E_{t-1}')$

E_t'

E_t''

Takeaway key notions

- Change detection $\leftrightarrow$ Looking for singularities in time series.
- Background representations:
    - Parameters of a single or multi-modal distribution.
    - Histogram of any distribution.
    - Sample of any distribution.
- Trade-off between computational cost (time, memory) / Representation complexity (number and length of statistics / value bins / modes / samples / ...)
- Space-time regularization: Markov fields, Mathematical Morphology,...

References

[Elga99] A. ELGAMMAL, D. HARDWOOD and L.S. DAVIS
Non-parametric Model for Background Subtraction
Proc. of ICCV '99 FRAME-RATE Workshop(1999)

[Stauf00] C. STAUFFER and C. GRIMSON
Learning patterns of activity using real-time tracking.
IEEE Trans. on PAMI 22(8), 747-757. (2000)

[Mittal04] A. MITTAL and N. PARAGIOS
Motion-based background subtraction using adaptive kernel density estimation.
IEEE CVPR'04

References

- **[Power02]** P. POWER and J. SCHONEES
  Understanding background mixture models for foreground segmentation.
  In: Imaging and Vision Computing New Zealand, Auckland, NZ (2002)

- **[Manza07a]** A. MANZANERA and J. RICHEFEU
  A new motion detection algorithm based on Sigma-Delta background estimation.
  Pattern Recognition Letters 28(3), 320-328. (2007)

- **[Manza07b]** A. MANZANERA
  Sigma-Delta Background Subtraction and the Zipf Law.
  Progress in Pattern Recognition, Image Analysis and Applications (CIARP'07) pp. 42-51.

References

- **[Wang07]** H. WANG and D. SUTER
  A consensus-based method for tracking: Modelling background scenario and foreground appearance
  Pattern Recognition 40(3), 1091-1105. (2007)

- **[Barnich09]** O. BARNICH and M. VAN DROGENBROECK
  ViBe: a powerful random technique to estimate the background in video sequences
  International Conference on Acoustics, Speech, and Signal Processing 945-948. (2009)

- **[Bouthémy93]** P. BOUTHÉMY and P. LALANDE
  Recovery of moving object masks in an image sequence using local spatiotemporal contextual information.
  Optical Engineering, 32(6):1205-1212, June 1993.