Background Subtraction

Jenn-Jier James Lien (連 震 杰)

Professor

Computer Science and Information Engineering

National Cheng Kung University

(O) (06) 2757575 ext. 62540

jjlien@csie.ncku.edu.tw

http://robotics.csie.ncku.edu.tw

Major Issues

1. **Parametric modeling Vs. non-parametric modeling**

2. **Gaussian Mixture Model (GMM)**

3. **Difference between AI and ML**

4. **1) How do you model the background?**

    2) How do you update the background parameters?

■ Artificial Intelligent:

➢ Learning once based on existed training database

➢ Posterior Prob. = Likelihood Prob. * ~~Priori Prob~~. / k
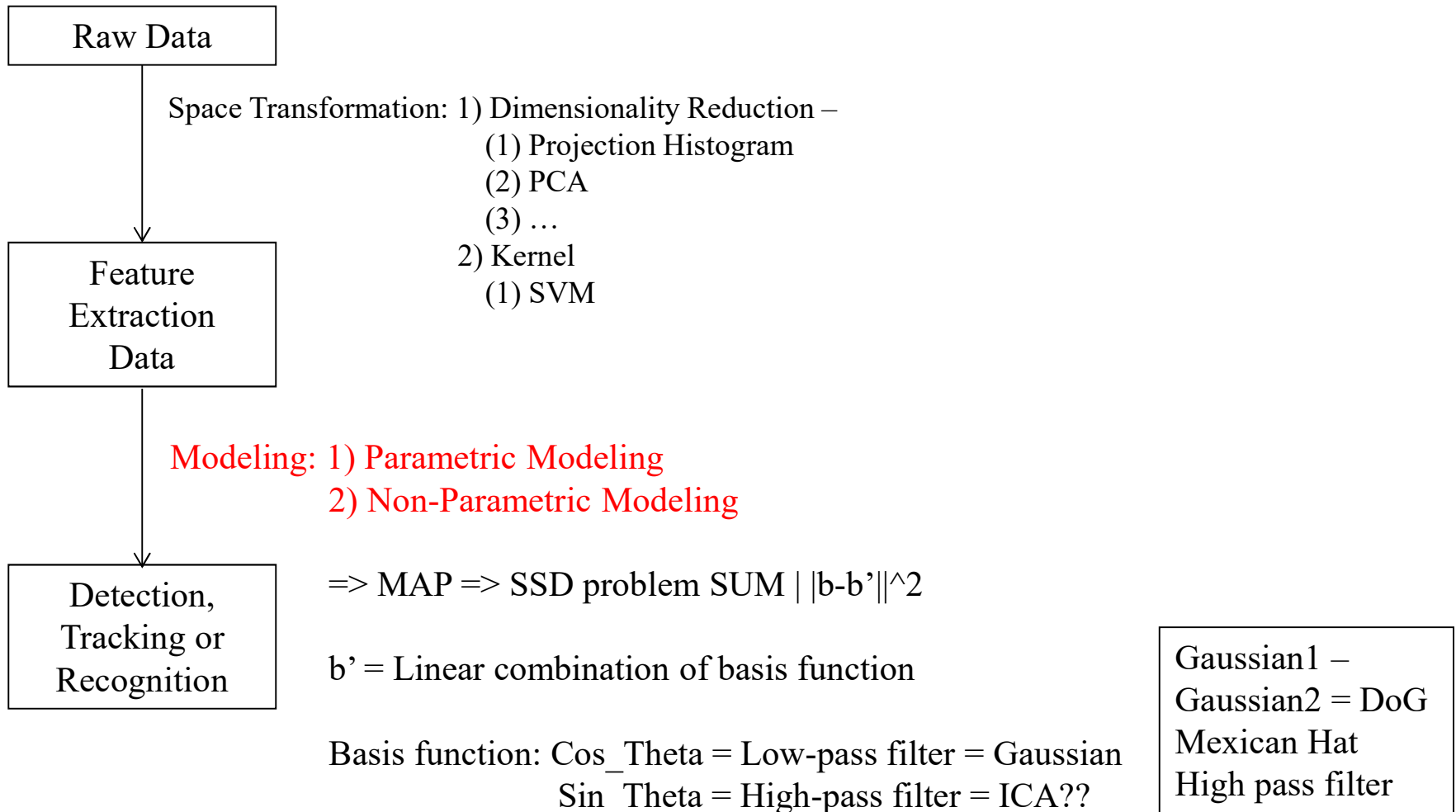
機器學習和人工智慧（Artificial Intelligence, AI）的差異：
-人工智慧: 希望「機器智能化」，而
-機器學習: 則是「讓機器具有學習能力」，也就是讓電腦擁有
「學著變聰明的能力」，概念上比人工智慧容易達到。

■ Machine Learning:

➢ Keep learning (incremental learning) after learning existed training database

● Information is captured via sensors

➢ Update learning database/knowledge

● How to add new data/knowledge?

● How to delete unwanted data/knowledge or reduce the affection by those old data/knowledge ?

➢ Posterior Prob. = Likelihood Prob. * Priori Prob. / k

About Modeling

Raw Data

Space Transformation: 1) Dimensionality Reduction –
        (1) Projection Histogram
        (2) PCA
        (3) …
      2) Kernel
        (1) SVM

Feature
Extraction
Data

Modeling: 1) Parametric Modeling
        2) Non-Parametric Modeling

Detection,
Tracking or
Recognition

=> MAP => SSD problem SUM $||b-b'||^2$

b' = Linear combination of basis function

Basis function: Cos_Theta = Low-pass filter = Gaussian
        Sin_Theta = High-pass filter = ICA??

Gaussian1 –
Gaussian2 = DoG
Mexican Hat
High pass filter

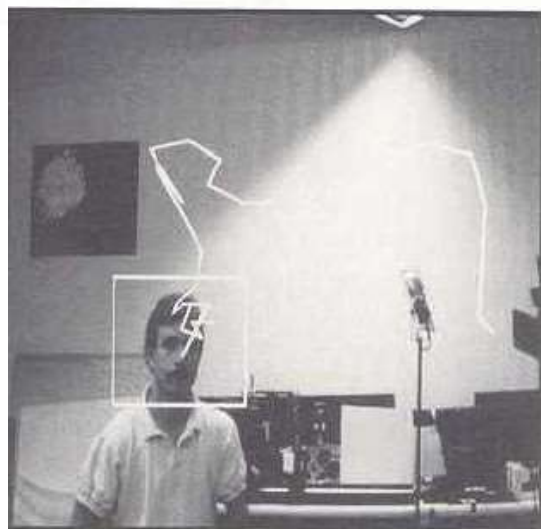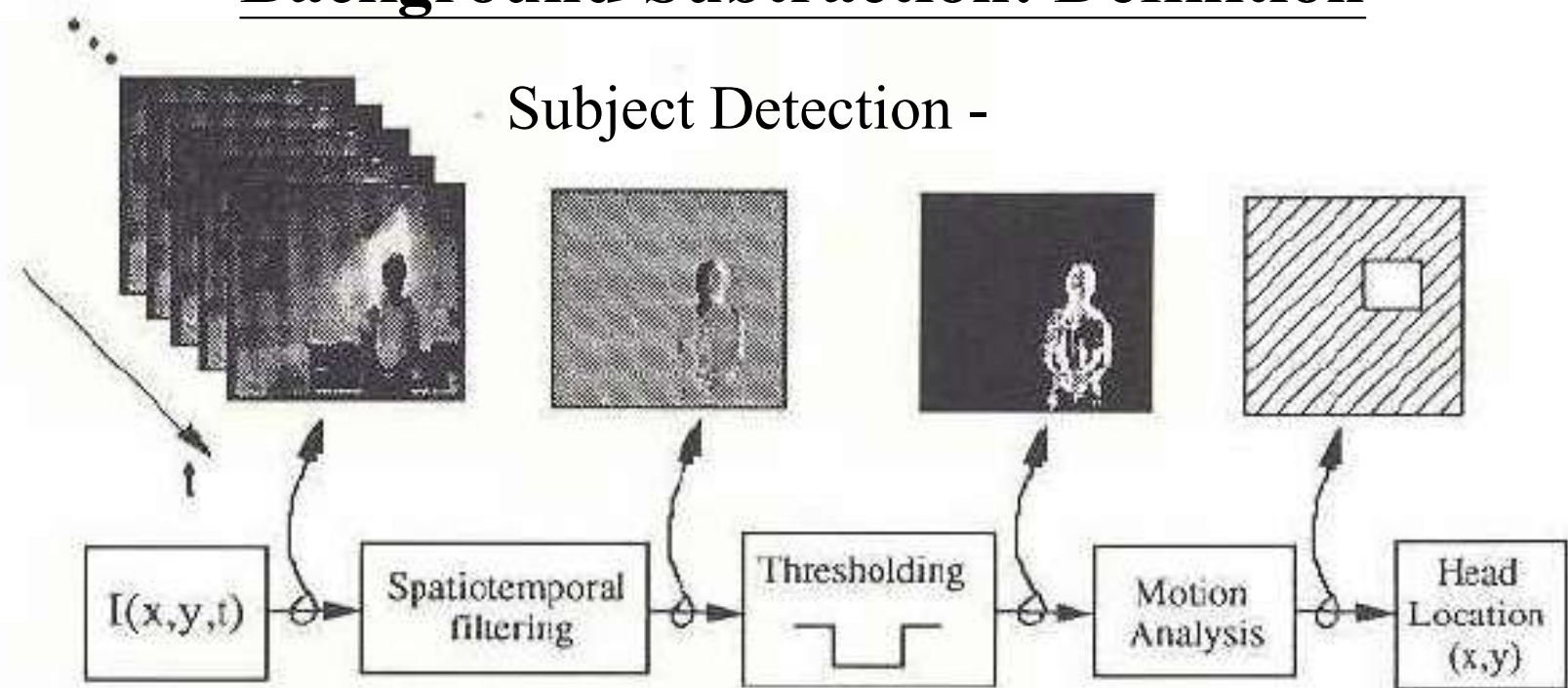# Modeling: Background Modeling

❑ **Modeling Process:**

    1) From analog data to digital data

    2) Analyze digital data and then define, quantize

    3) Programming

❑ **Two types of modeling:**

    1) Parameter Modeling: Gaussian Model N(m, var)

    2) Non-Parameter Modeling:

Background Subtraction: Definition

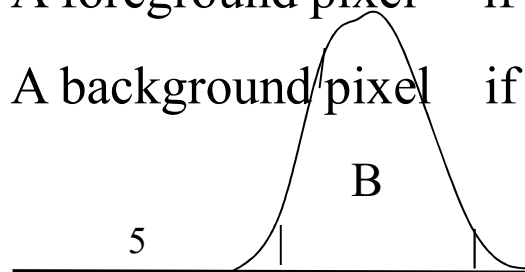Subject Detection -

$$B = 1/N \; \text{Sum} \; | \; I(x,y,t) - I(x,y,N) \; | \; , \quad N=1,2\ldots$$

A foreground pixel   if B >   threshold

A background pixel   if B <= threshold

B

Jenn-Jier James Lien

Time Varying Image Analysis:
Object/Subject Detection in the Spatio-Temporal Domain

- ❑ **Motion Detection**
  - ➢ Background Subtraction
  - ➢ Need Color ?
- ❑ **Motion Estimation**
  - ➢ Optical Flow
- ❑ **Egomotion and Structure From Motion**

The Problems

- Visual surveillance

  - stationary camera watches a workspace -find moving objects and alert an operator

  - moving camera navigates a workspace - find moving objects and alert an operator

- Image coding $\longrightarrow$ MPEG, H.264

  - use image motion to perform more efficient coding of images

- Navigation

  - camera moves through the world - estimate its trajectory

    » use this to remove unwanted jitter from image sequence - image stabilization and mosaicking

    » use this to control the movement of a robot through the world

❑ **Background modeling**

➢ One image Vs. image sequence

➢ Spatial domain and temporal domain

Motion Detection

- ■ Frame differencing

  - − subtract, on a pixel by pixel basis, consecutive frames in a motion sequence

  - − high differences indicate change between the frames due to either motion or changes in illumination

- ■ Problems

  - − noise in images can give high differences where there is no motion

    - » compare neighborhoods rather than points

  - − as objects move, their homogeneous interiors don't result in changing image intensities over short time periods

    - » motion detected only at boundaries

    - » requires subsequent grouping of moving pixels into objects

- ■ Background subtraction
  - – create an image of the <u>stationary background</u> by averaging a long sequence
    - » for any pixel, most measurements will be from the background
    - » computing the median measurements, for example, at each pixel, will with high probability assign that pixel the true background intensity - fixed threshold on differencing used to find "foreground" pixels
    - » can also compute a distribution of background pixels by fitting a mixture of Gaussians to set of intensities and assuming large population is the background - adaptive thresholding to find foreground pixels
  - – difference a frame from the known background frame
    - » even for interior points of homogeneous objects, likely to detect a difference
    - » this will also detect objects that are stationary but different from the background
    - » typical algorithm used in surveillance systems
- ■ Motion detection algorithms such as these only work if the camera is stationary and objects are moving against a fixed background
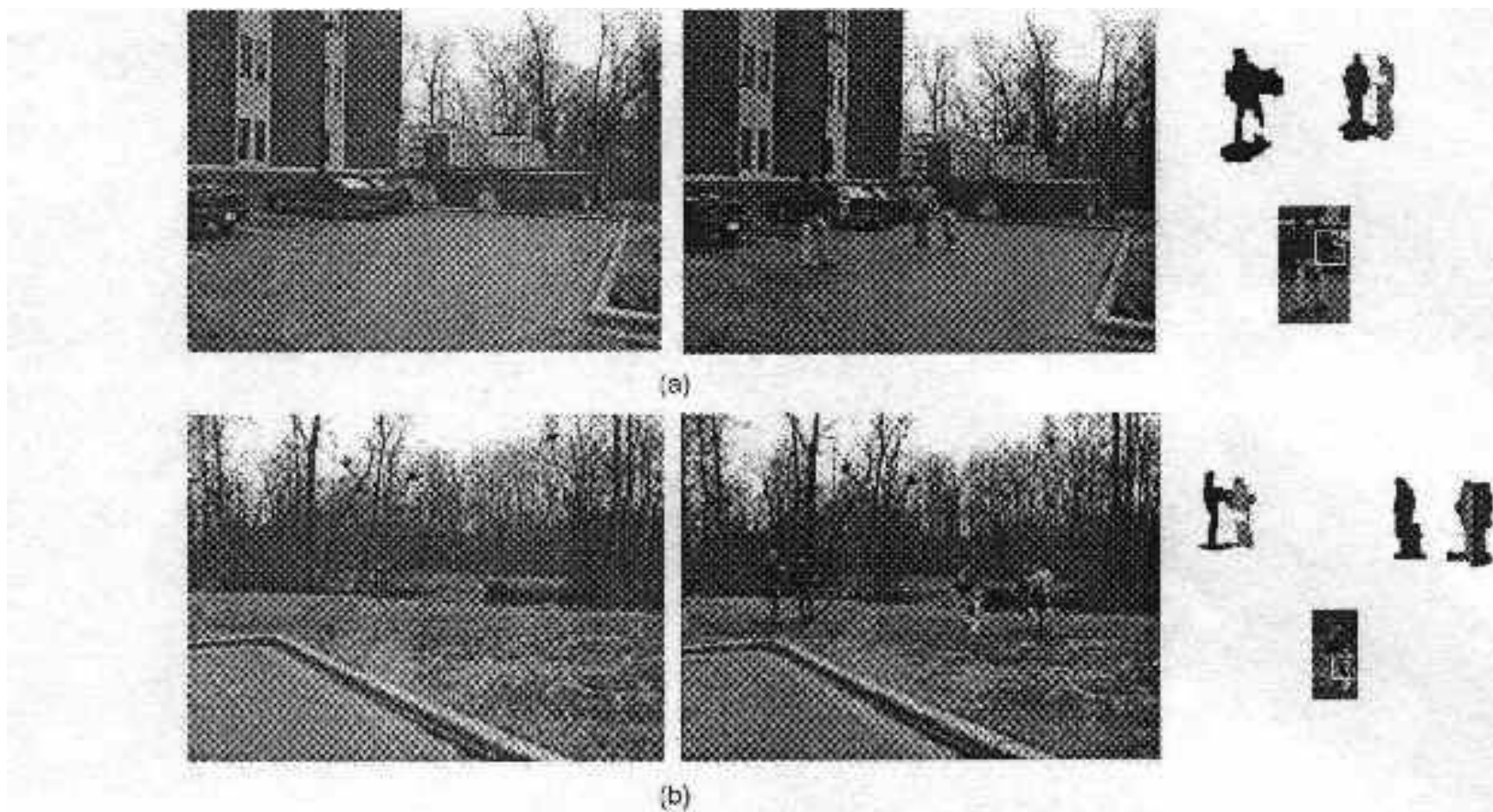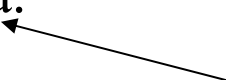
Fig. 1. Example of detection of people.

Objectives

•Robust detection of moving targets in complex
outdoor situations from a static camera.

How about active
camera ?

1) Modeling

- High sensitivity detection of real targets.

- Low false alarm rates

2) Updating

- Adaptation to change in the scene: fast or slow (updating).

>Work with gray level / color imagery.

>Suppress shadows from detection.

•Applications : General outdoor video surveillance systems
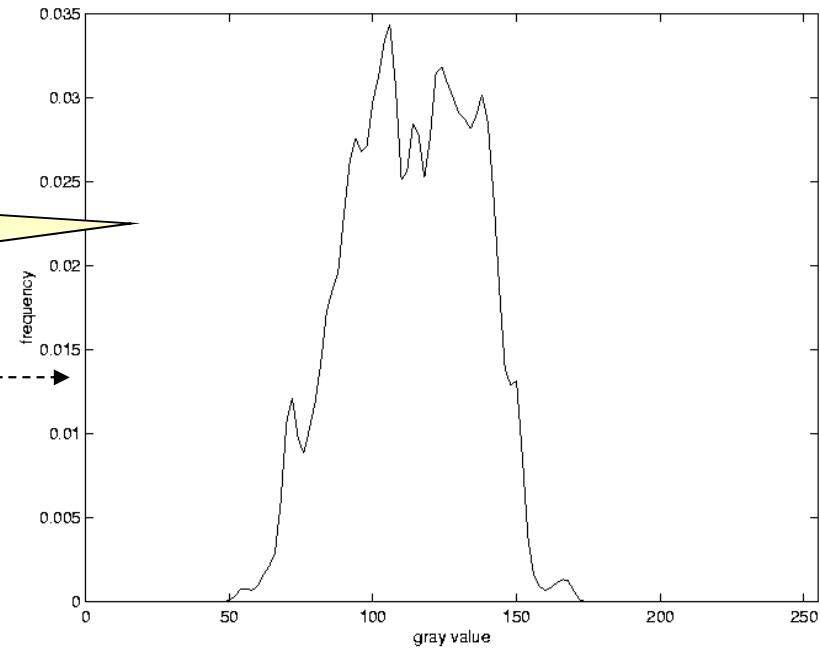
Difficulties with Outdoor Scenes

- Background is not completely static :
  - Tree branches & bushes movement depends on the wind.
  - Change in lightening conditions: slow or fast
- Pixel intensity varies significantly over time. One pixel can be image of the sky at one frame, tree leaf at another frame, tree branch on a third frame and some mixture subsequently.

How can we model
this signal sequence?

Intensity Value of a
pixel over a 900
frames - (30
seconds)
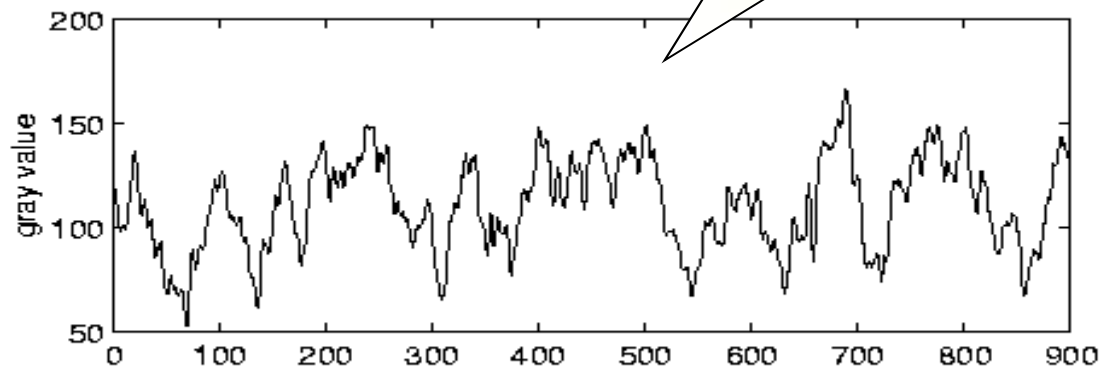
Motivations

Intensity histogram of a
pixel over 30 seconds
(900 frames)

- Intensity distribution is changing
  dramatically over short periods of
  time.

- Modeling intensity variation over
  long period leads to wide
  distributions which results in poor
  detection.

  easily confuse

- Using more "short-term"
  distributions will allow better
  detection sensitivity.

9 histograms each
over 100 frames
(3 1/3 seconds)

Previous
page with
temporal
(900 frames)
distribution
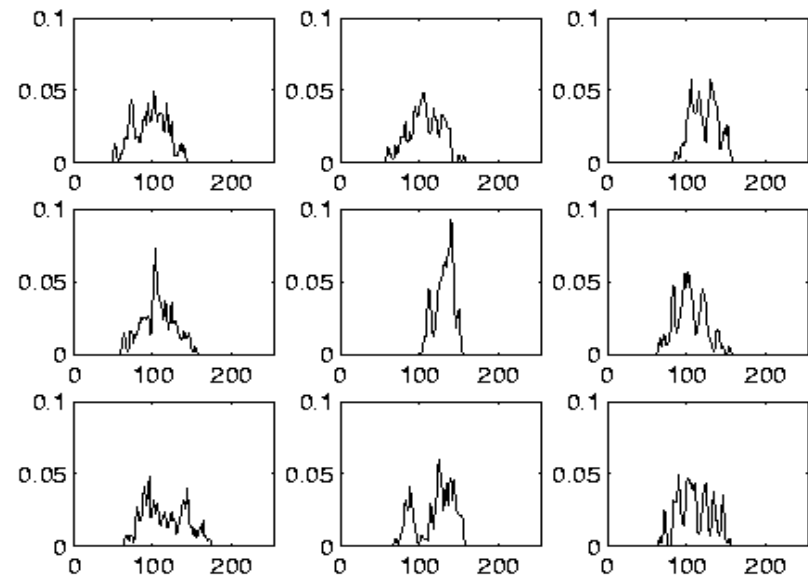
Background Subtraction Method I:

Parametric Model
for
Background Subtraction:

Gaussian Mixture Model (GMM)
(or Mixture of Gaussian Model)

Background Subtraction Method I:
A Mixture of *K* Gaussian Distributions (GMM)

Likelihood
Prob:

K=3~5 clusters using VQ (vector quantization)

$$P_r(x_t) = \sum_{j=1}^{K} \frac{w_j}{2\pi^{\frac{d}{2}} \left| \sum_j \right|^{\frac{1}{2}}} e^{-\frac{1}{2}(x_t - \mu_j)^T \Sigma_j^{-1}(x_t - \mu_j)}$$

w_j = # samples in each cluster j/

Cluster ordered by $\frac{w_j}{\sigma_j^2}$ (distribution density?)

Total samples; Priori Prob. (as weight)

d = r, g, b channels

$$B = \arg\min \left(\frac{\sum_{j=1}^{b} w_j}{\sum_{j=1}^{K} w_j} > T = 98.9\% \right)$$

If $\sigma_j > 2.5$, then foreground, otherwise background

Gaussian Model (Probability): m+-Lo (66%?), m+-2Lo (95%?), m+-3Lo (99%)

CS Using EM (Expectation-Maximization) to solve $\{w_j, u_j$ and $\sigma_j\}$

Background Subtraction Method II:

Non-parametric Model
for
Background Subtraction

II.1 Basic Background Model (in Temporal Domain)

- Capture very recent history about the scene.

- Continuously updating this history to capture fast changes.

- Pixel Model : N intensity samples/images $x_1, x_2, ..., x_N$ taken over time window W.

  first in first out $\longleftarrow [x_1, x_2, ..., x_N] \longleftarrow$

- Estimate the probability that a new observed intensity comes from the same distribution using kernel K_h :

  x_t

$$\Pr(x_t) = \frac{1}{N} \sum_{i=1}^{N} K_h (x_t - x_i)$$

Parzon Window??

$$\Pr(x_t) = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{(2\pi)^{\frac{d}{2}} \left| \sum \right|^{\frac{1}{2}}} e^{-\frac{1}{2}(x_t - x_i)^T \sum^{-1}(x_t - x_i)}$$

$\Pr(x_t(x,y))$

Mathematics is this way by considering entire image x_t, but reality it is based on pixel $x_t(x,y)$ over N images, that is, modeling each pixel (x,y) corresponding to pixel probability $\Pr(x_t(x,y))$ over N images for the same position pixel

$$\Pr(x_t) = \frac{1}{N}\sum_{i=1}^{N} K_h(x_t - x_i)$$

- We use Normal kernel function, $K_h = N(0, \Sigma)$ $d=3$

- Σ represents the kernel function bandwidth.

$$\Sigma = \begin{pmatrix} \sigma_1^2 & 0 & 0 \\ 0 & \sigma_2^2 & 0 \\ 0 & 0 & \sigma_3^2 \end{pmatrix}$$

$$\Pr(x_t) = \frac{1}{N}\sum_{i=1}^{N}\prod_{j=1}^{d}\frac{1}{\sqrt{2\pi\sigma_j^2}}e^{-\frac{1}{2}\frac{(x_{t_j}-x_{i_j})^2}{\sigma_j^2}}$$

For grayvalue image, $d=1$

A foreground pixel   if $\Pr(x_t) <$ threshold ← No updating

A background pixel   if $\Pr(x_t) >=$ threshold ← Updated

jump.mlv

- Threshold the estimated probability to obtain the foreground.

Original Image   Estimated Probability Image   Detected Foreground

Kernel Width Estimation

- Adaptively estimate suitable kernel function bandwidth for each pixel and for each color channel

  - Objective: Measure variation in pixel intensity when the pixel is a projection of the same object.

  - How: Use median, *m,* of absolute deviation between consecutive intensity values (in time) $|x_i - x_{i+1}|$ to estimate kernel bandwidth.

$$x \approx N(\mu, \sigma^2) \Rightarrow x_i - x_{i+1} \approx N(0, 2\sigma^2) \Rightarrow \sigma = \frac{m}{0.68\sqrt{2}}$$

$$\mu \pm 2\sigma = m + 2\sigma \quad \longleftarrow \quad 95\%$$

II.2 Suppression of False Detection (in Spatial Domain)

- Suppress detected pixels that are likely to be displaced from a nearby point (High Pixel Displacement Probability) as a result of:

    - Movements in the background (affected by clouds or wind).

    - Camera displacement.

1) **Pixel Displacement Probability** :
$$P_N(x_t) = \max_{y \in N(x)} \Pr(x_t | B_y)$$

the background
sample for pixel y

Circular neighborhood, diameter = 5 pixels

Maximum probability that the observed intensity value x_t belongs to the background distribution of some point in the neighborhood $N(x)$.

2) **Constraint: Component displacement probability**
$$P_C = \prod_{x \in C} P_N(x)$$

The whole detected foreground object (connected component) must have moved from a nearby location.

compare with Bkgd neighbors

1) Pixel Displacement Probability :
$$P_N(x_t) = \max_{y \in N(x)} \Pr(x_t | B_y)$$

2) Component displacement probability:
$$P_C = \prod_{x \in C} P_N(x)$$

include obj. neighbors

- For a connected component corresponding to a real target, the probability that this component has displaced from the background will be very small. So, a detected pixel x will be considered to be a part of the background only if

Background: $(P_N(x) > threshold\,1) \wedge (P_C(x) > threshold\,2)$

Easily cause the overkill to foreground pixels

For connected components (CC),
if all are background pixels, then Pc(x) is high
if some are foreground pixels, then Pc(x) is low,
then cannot kill this foreground pixels

The camera has been slightly displaced during this time interval, so we see many false detection along high contrast edges.

First Stage

(No background update is used in this example- the background is several seconds old)

Suppressing detected pixels with high displacement probability. (cause overkill)
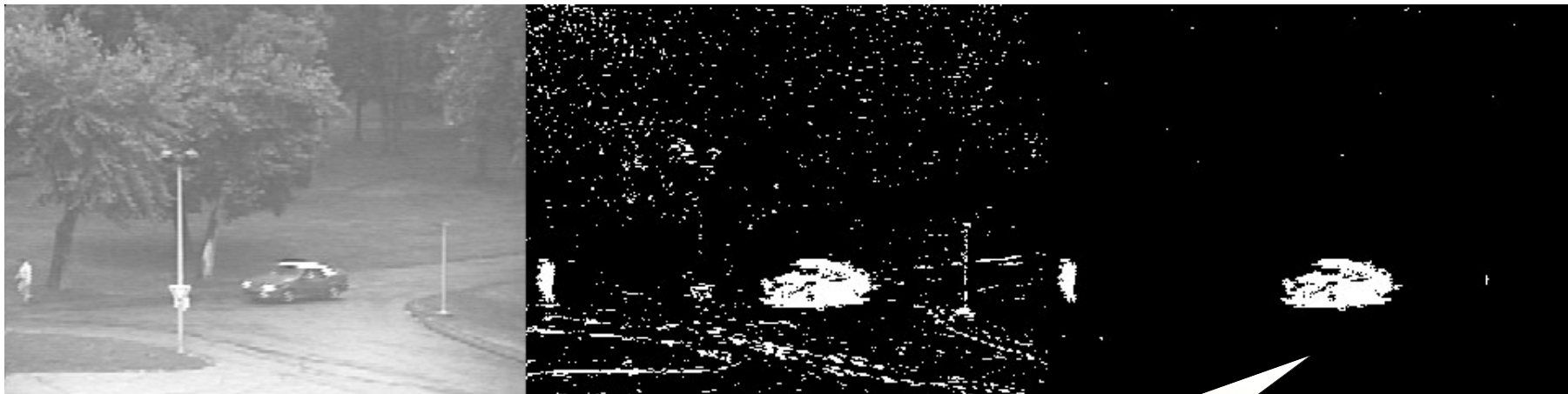
Add component displacement probability constraint to avoid too many deletion.

23

Jenn-Jier James Lien

As a result of the wind load, the camera is shaking slightly which results in a lot of clustered false detections especially on the edges.

Camera jitter caused clustered false detections

shaking.avi

Result after applying pixel & component displacement probability constraints

II.3 Shadow Suppression (in Spatial Domain)

- Color information is useful for suppressing shadows from detection by separating chromaticity information from lightness information.

- Using the chromaticity coordinates in detection has the advantage of being more insensitive the small changes in illumination that are due to shadows.

- Using the chromaticity has the disadvantage of losing lightness information.

- Using chromaticity coordinates ($r = \frac{R}{R+G+B}, g = \frac{G}{R+G+B}$) for probability estimation.

- Applying probability estimation in the (r,g) space by using the constrained samples.

$$r = \frac{R}{R+G+B}, g = \frac{G}{R+G+B}, b = \frac{B}{R+G+B} \quad where: r + g + b = 1$$

R=Sr
G=Sg
B=Sb

- Using lightness variable ($s=R+G+B$) as a lightness measure to constraint pixel history to "relevant" samples only. (Ps: $s=(R+G+B)/3$ as *gray value*)
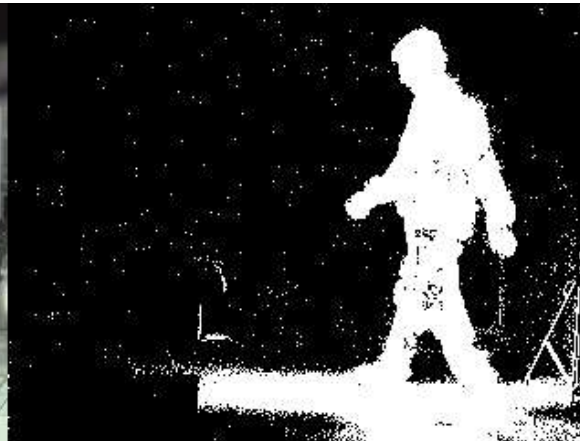
Original background pixel (before frame t): $x_i = <r_i, g_i, s_i>$

Background pixel covered by shadow in frame t: $x_t = <r_t, g_t, s_t>$

darker

Background pixel: $B = \{x_i \mid (x_i \in A) \wedge (0 < \alpha = 10^{-6} \leq \dfrac{s_t}{s_i} \leq \beta < 1)\}$

A: The sample values representing the background for a certain pixel
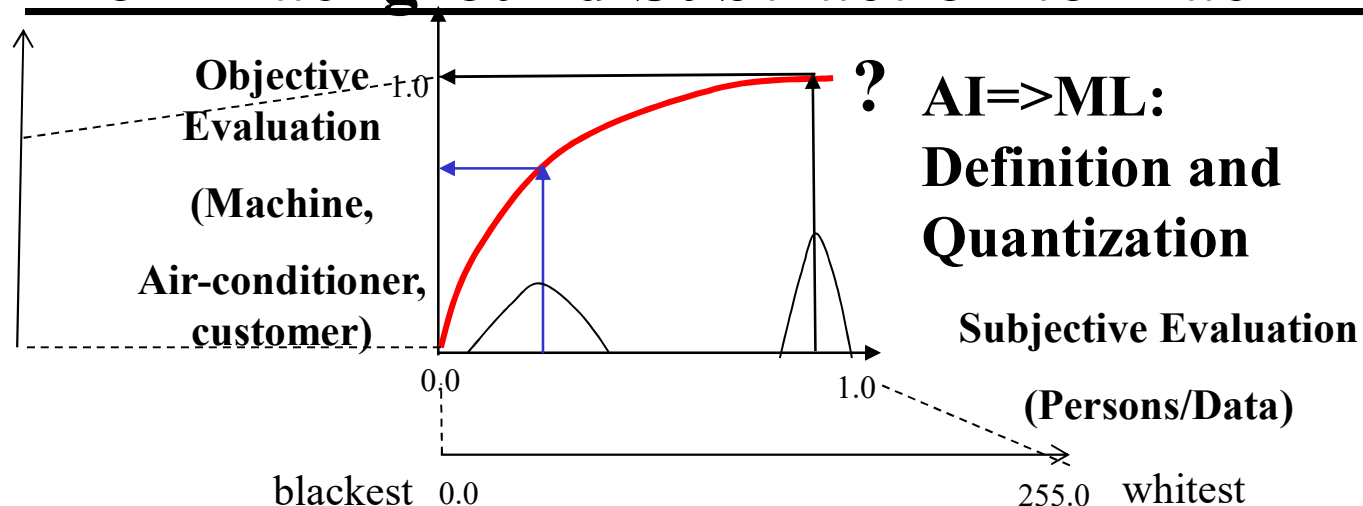
?

Ismail.avi

Detection using (R,G,B)   Detection using (r,g) constrained by s

III.4 Updating the Background

- Objective: keep recent, representative samples of pixel intensity.

- Sample new intensity values periodically.
    - Sample pairs of intensity values (Consecutive frames)
    - Discard old samples (First-in First-out)
    - Select a new sample randomly from each time interval.

- Update issues :   ⟵———   New AI: ML => Keep learning
    - How fast to update ?
    - Where/which one to update ?

From Background Subtraction to Machine Learning

Objective Evaluation (Machine, Air-conditioner, customer)

1.0

? **AI=>ML: Definition and Quantization**

Example: Histogram equalization

0.0

1.0

Subjective Evaluation (Persons/Data)

blackest 0.0

255.0 whitest

■ Artificial Intelligence:

➢ Learning once based on existed training database

➢ Posterior Prob. = Likelihood Prob. * Priori Prob. / k
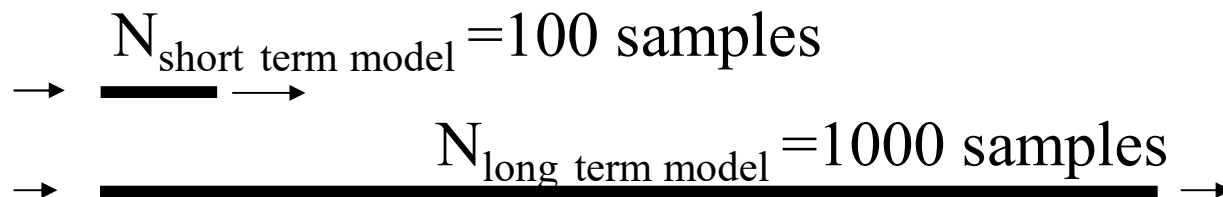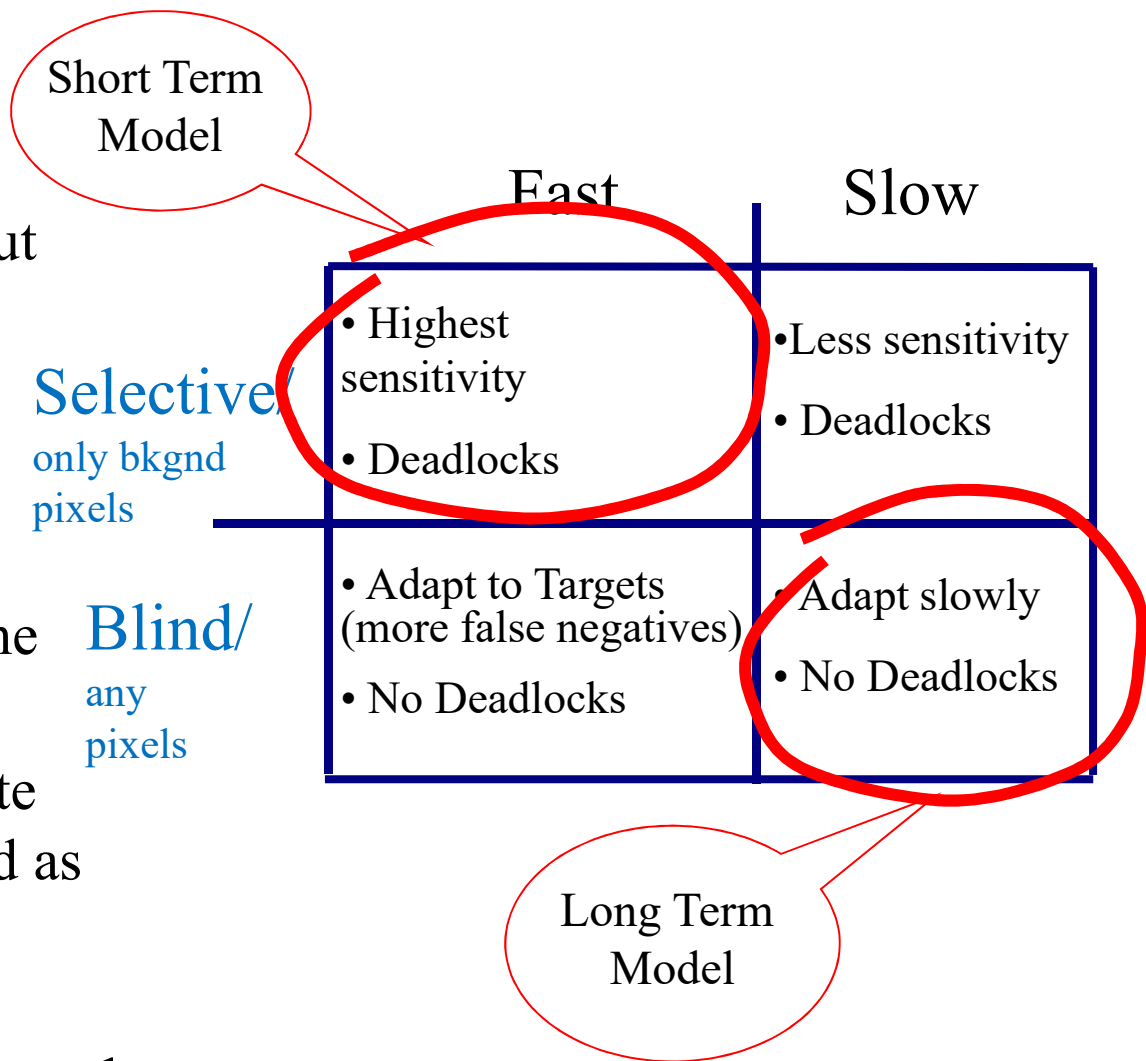
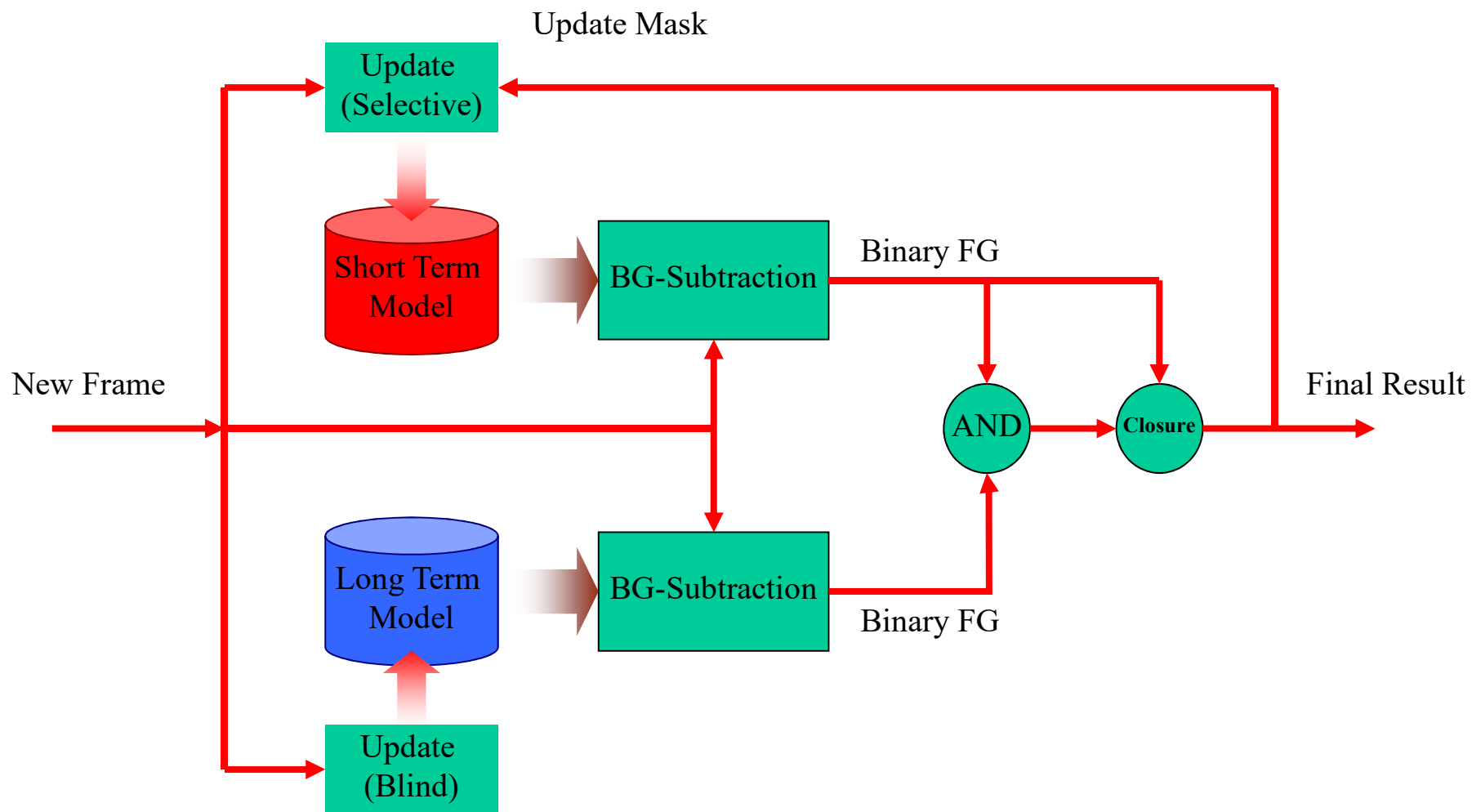機器學習和人工智慧（Artificial Intelligence, AI）的差異：
-人工智慧: 希望「機器智能化」，而
-機器學習: 則是「讓機器具有學習能力」，也就是讓電腦擁有「學著變聰明的能力」，概念上比人工智慧容易達到。

■ Machine Learning:

➢ Keep learning (incremental learning) after learning existed training database

● Information is captured via sensors

➢ Update learning database/knowledge

● How to add new data/knowledge?

● How to delete unwanted data/knowledge or reduce the affection by those old data/knowledge ?

➢ Posterior Prob. = Likelihood Prob. * Priori Prob. / k

28

- How fast to update ?
  - Fast update : what about targets ?
  - Slow update : Wider distributions !
- *Where to update ?*
  - Blind update: update the model for all pixels.
  - Selective update: update only for pixel classified as background.

Short Term Model

Fast

Slow

Selective/
only bkgnd
pixels

	Fast	Slow
Selective/ only bkgnd pixels	• Highest sensitivity • Deadlocks	•Less sensitivity • Deadlocks
Blind/ any pixels	• Adapt to Targets (more false negatives) • No Deadlocks	• Adapt slowly • No Deadlocks

Blind/
any
pixels

Long Term Model

$N_{short\ term\ model} = 100$ samples

$N_{long\ term\ model} = 1000$ samples

Summary

1. Basic Background Model (in Temporal Domain):

$$\Pr(x_t) = \frac{1}{N}\sum_{i=1}^{N}\prod_{j=1}^{d}\frac{1}{\sqrt{2\pi\sigma_j^2}}\, e^{-\frac{1}{2}\frac{(x_{t_j}-x_{i_j})^2}{\sigma_j^2}}$$

A foreground pixel   if $\Pr(x_t) <$ threshold

A background pixel   if $\Pr(x_t) >=$ threshold

2. Suppression of False Detection (in Spatial Domain):

1) **Pixel displacement probability:** $P_N(x_t) = \max_{y \in N(x)} \Pr(x_t | B_y)$

2) **Component displacement probability:** $P_C = \prod_{x \in C} P_N(x)$

$$\left(P_N(x) > threshold\,1\right) \wedge \left(P_C(x) > threshold\,2\right) \; => \text{background pixel}$$

3. Shadow Suppression (in Spatial Domain)

Original background pixel (before frame t): $x_i = <r_i, g_i, s_i>$

Background pixel covered by shadow in frame t: $x_t = <r_t, g_t, s_t>$

Background pixel: $B = \{x_i \mid (x_i \in A) \wedge (0 \le \alpha = 10^{-6} \le \dfrac{s_t}{s_i} \le \beta \le 1)\}$

A: The sample values representing the background for a certain pixel
?

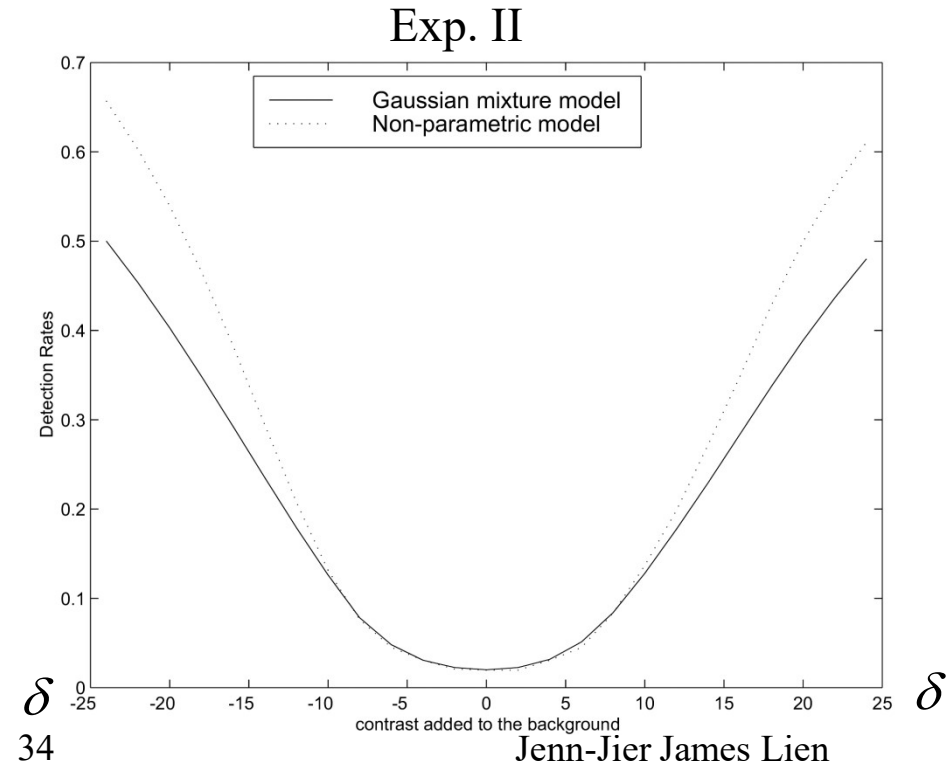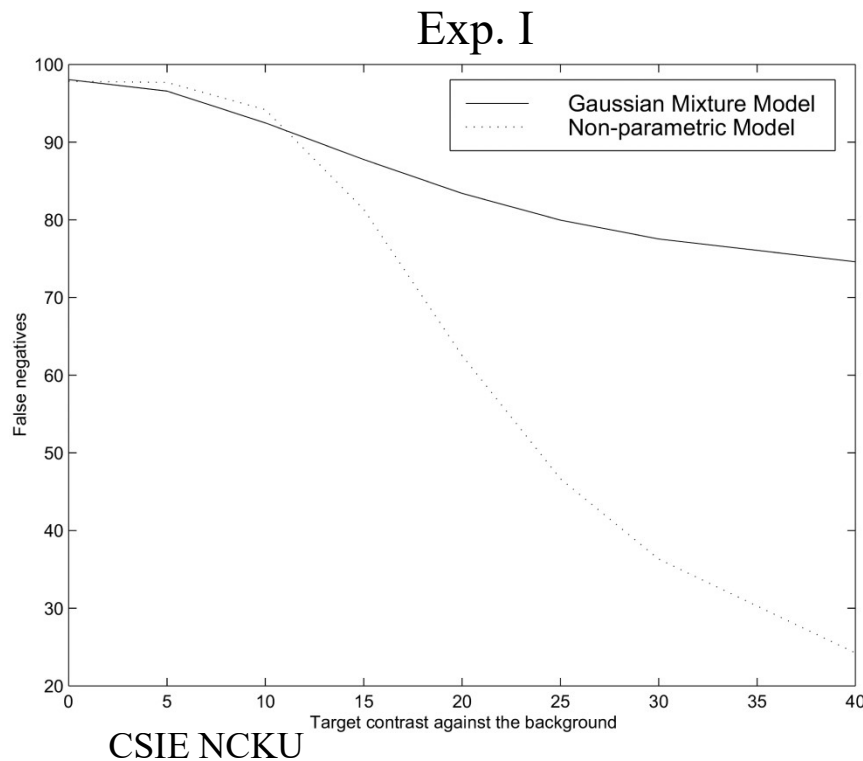## 4. Background Updating Using Short Term and Long Term Models

Evaluation

•Compare results to explicit mixture of Gaussian model.

•Measure the sensitivity to detect synthetic moving target with low contrast against the background.

• Exp I: How the false negative rate is affected by target presence in the scene.

•Exp II: Detection rate for low contrast targets without updating the model (target has no effect on the model)

demo2.mlv

- Synthetic Target: moving disk of radius 10 pixels with intensity $x_t + \delta$. Adjust model parameters to achieve 2% false positive rates.
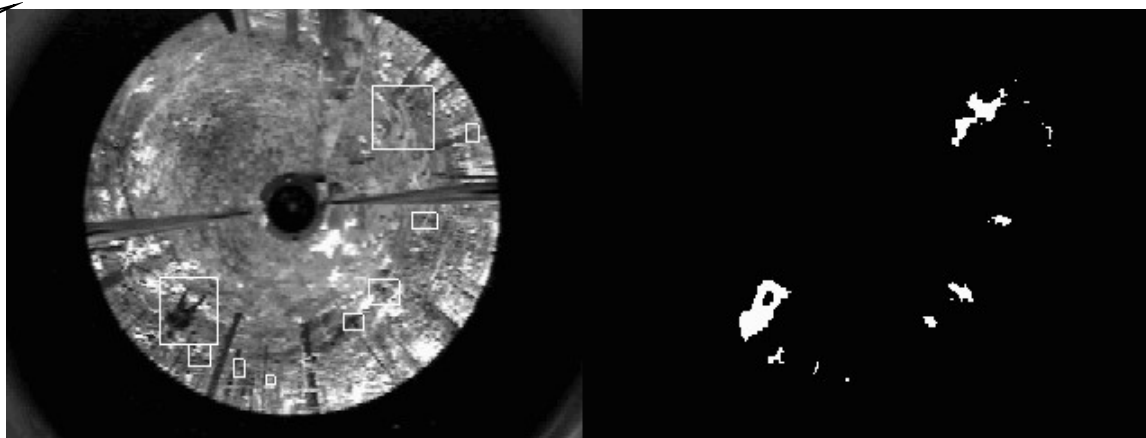
Scene used in evaluation experiments.

Exp. I

False negatives vs. Target contrast against the background
- Gaussian Mixture Model
- Non-parametric Model

δ

Exp. II

Detection Rates vs. contrast added to the background
- Gaussian mixture model
- Non-parametric model

δ

Results

demo_color.mlv

lehi4.mlv

Detection results for Omni-directional camera

Detection result for a rainy day.

rain.mlv

Implementation Issues

- Use Pre-calculated lookup tables for the kernel functions.

- Evaluate partial probability estimations only: Stop kernel calculation when the probability surpasses the required threshold (most image pixels are background pixels.)

- The implementation of the approach runs at 15-20 frames per second on a 400 MHz Pentium processors for 320x240 gray scale image frames and a background model of 50 (how about N=100 ?) samples/pixel for short term model and 1000 samples/pixel for long term model.

Background Subtraction Method III: W4
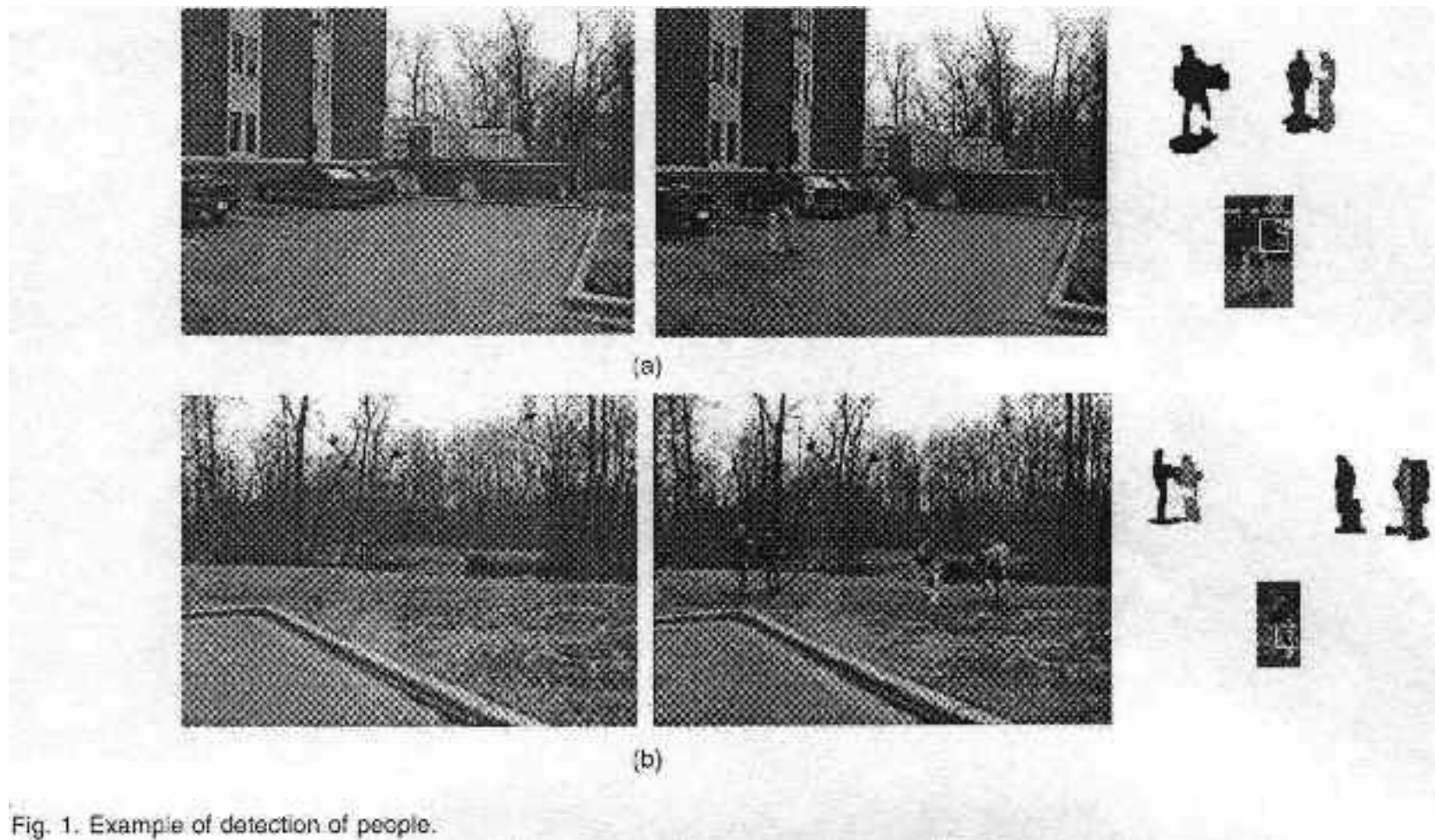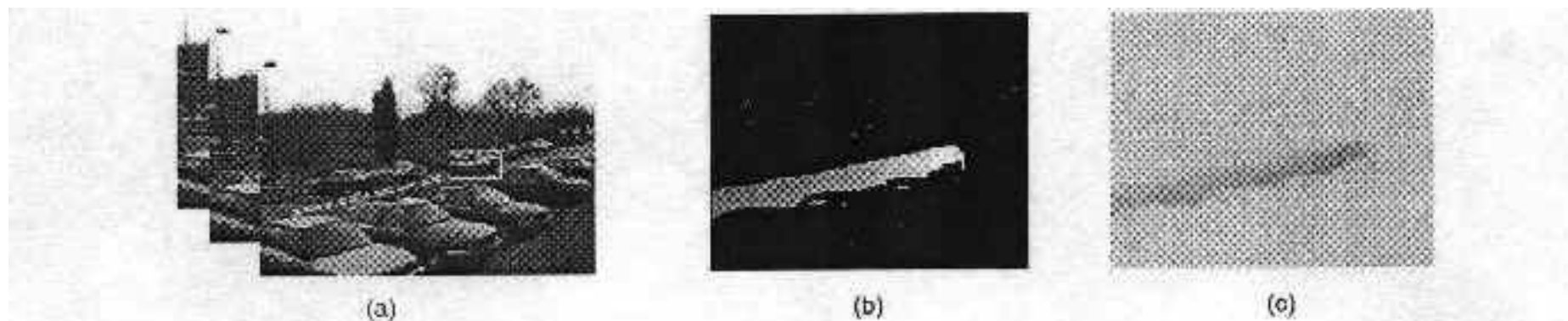

Fig. 1. Example of detection of people.

Fig. 3. An example of change-map used in background model computation: (a) input sequence, (b) motion history map, and (c) detection map.
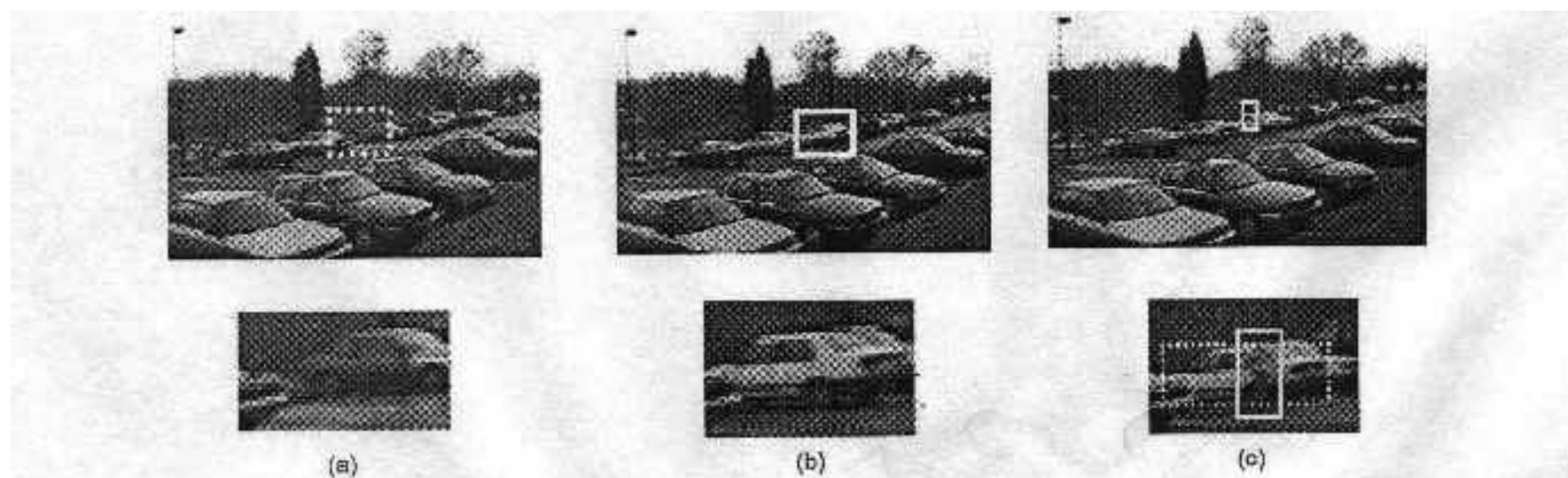

Fig. 4. A car which has been parked for a long time is added to background models (a) and (b), so the person getting off the car is detected (c).

III.1. Learning Initial Background Model

V: An array containing N consecutive images,

N=600~1200 frames (30 frames/sec).

$V^Z(x)$: The intensity of a pixel location x in the zth image of V.

d: As variance

$$\begin{bmatrix} m(x) \\ n(x) \\ d(x) \end{bmatrix} = \begin{bmatrix} \min_z \{V^z(x)\} \\ \max_z \{V^z(x)\} \\ \max_z \{|V^z(x) - V^{z-1}(x)|\} \end{bmatrix}$$

J: Feel it is two-time training process.

$$where \quad |V^z(x) - \lambda(x)| < 2*\sigma(x)$$

$$\mu \pm 2\sigma \quad \longleftarrow \quad 95\%$$

Standard deviation and **median value** of intensities at pixel location x in all image in V

Here, $V^Z(x)$: is classified as stationary pixels = background pixels.

Updating Background Model Parameters: Counter

A detection support map (gS):

$$gS(x,t) = \begin{cases} gS(x,t-1)+1 & \text{if } x \text{ is a background pixel} \\ gS(x,t-1) & \text{if } x \text{ is a foreground pixel} \end{cases}$$
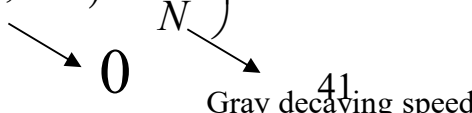
A motion support map (mS):

$$mS(x,t) = \begin{cases} mS(x,t-1)+1 & \text{if } M(x,t)=1 \text{ a moving pixel} \\ mS(x,t-1) & \text{if } M(x,t)=0 \text{ a stationary pixel} \end{cases}$$

$$\text{where } M(x,t) = \begin{cases} 1 & \begin{array}{l} \text{if } \left(\left|I(x,t-1)-I(x,t)\right| > 2*\sigma\right) \wedge \\ \left(\left|I(x,t-1)-I(x,t-2)\right| > 2*\sigma\right) \end{array} \\ 0 & \text{otherwise,} \end{cases}$$

A change history map (hS): represent the elapsed time (in frames) since the last time that the pixel was classified as a foreground pixel.

$$hS(x,t) = \begin{cases} 255 & \text{if } x \text{ is a foreground pixel} \\ \left(hS(x,t-1)-\dfrac{255}{N}\right) & \text{otherwise,} \end{cases}$$

0   Gray decaying speed

b: background pixel

f: foreground pixel

c: the background model parameters currently being used

[m(x),n(x),d(x)]: the new background model parameters

$$[m(x), n(x), d(x)] = \begin{cases} \left[m^b(x), n^b(x), d^b(x)\right] & if\,(gS(x) > k*N) \quad (pixel - base) \\ \left[m^f(x), n^f(x), d^f(x)\right] & if\,(gS(x) < k*N \wedge mS(x) < r*N) \\ & \qquad (object\text{-}base) \\ \left[m^c(x), n^c(x), d^c(x)\right] & otherwise \end{cases}$$

?

where k=0.8 and r=0.1

III.2.1 Foreground Region Detection in Temporal Domain

d: As variance

$$\left[m^b(x), n^b(x), d^b(x)\right]$$

$$B(x) = \begin{cases} 0\ background & if\ \left(\mid I^t(x) - m(x)\mid < kd_u\right) \vee \left(\mid I^t(x) - n(x)\mid < kd_u\right) \\ 1\ foreground & otherwise, \end{cases}$$

where k=2
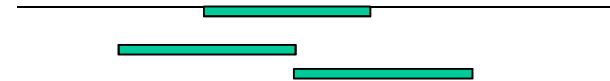
III.2.2 Foreground Region Detection in Spatial Domain

❑ **Procedure of segmenting foreground objects from background pixels:**

1) Thresholding (background subtraction)

2) Morphological Filtering/Dilation and Erosion

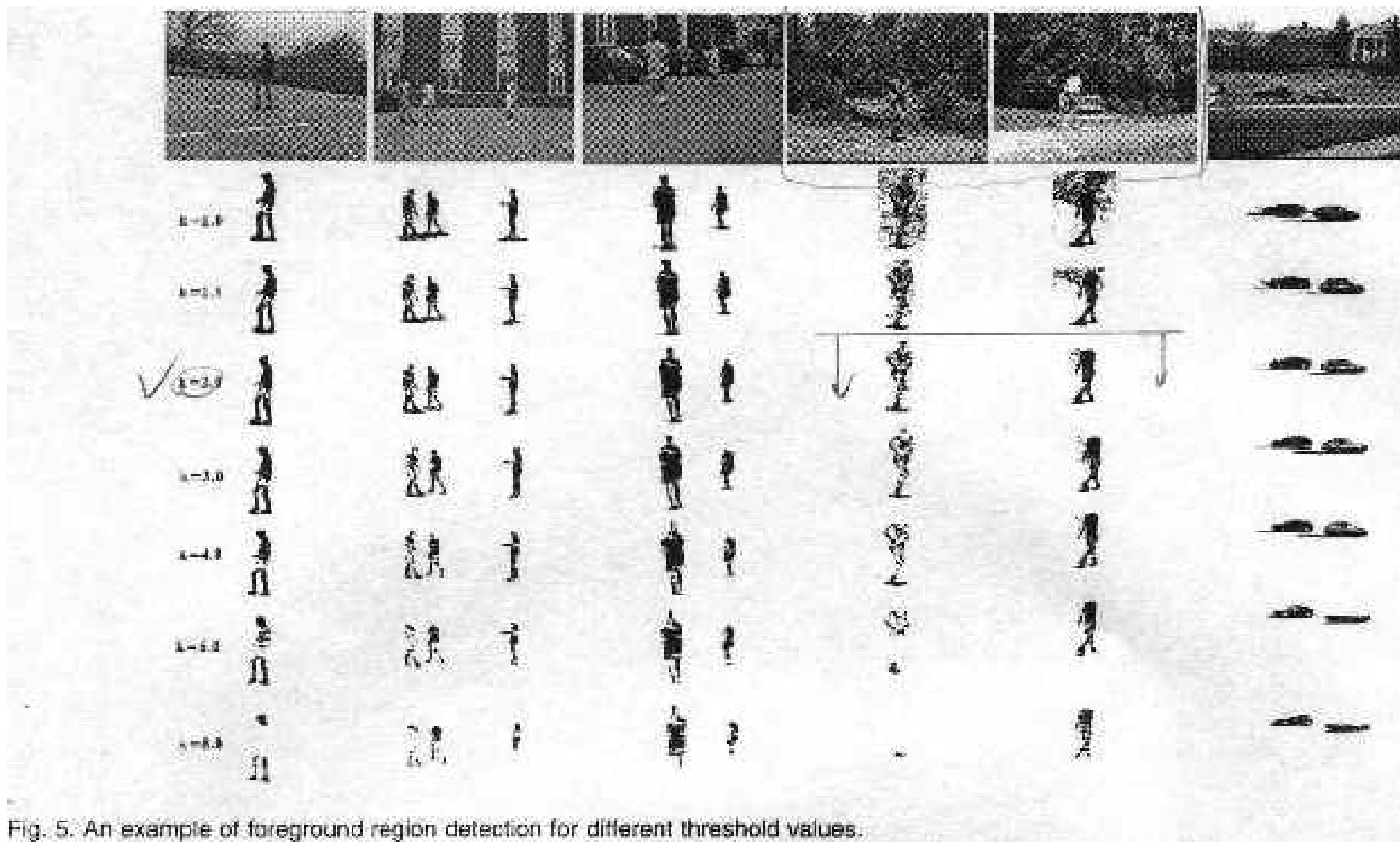3) Noise cleaning/CC Labeling

4) Object detection

Fig. 5. An example of foreground region detection for different threshold values.

Table: True detection rate for different background scenes which have different intensity variations as previous figure

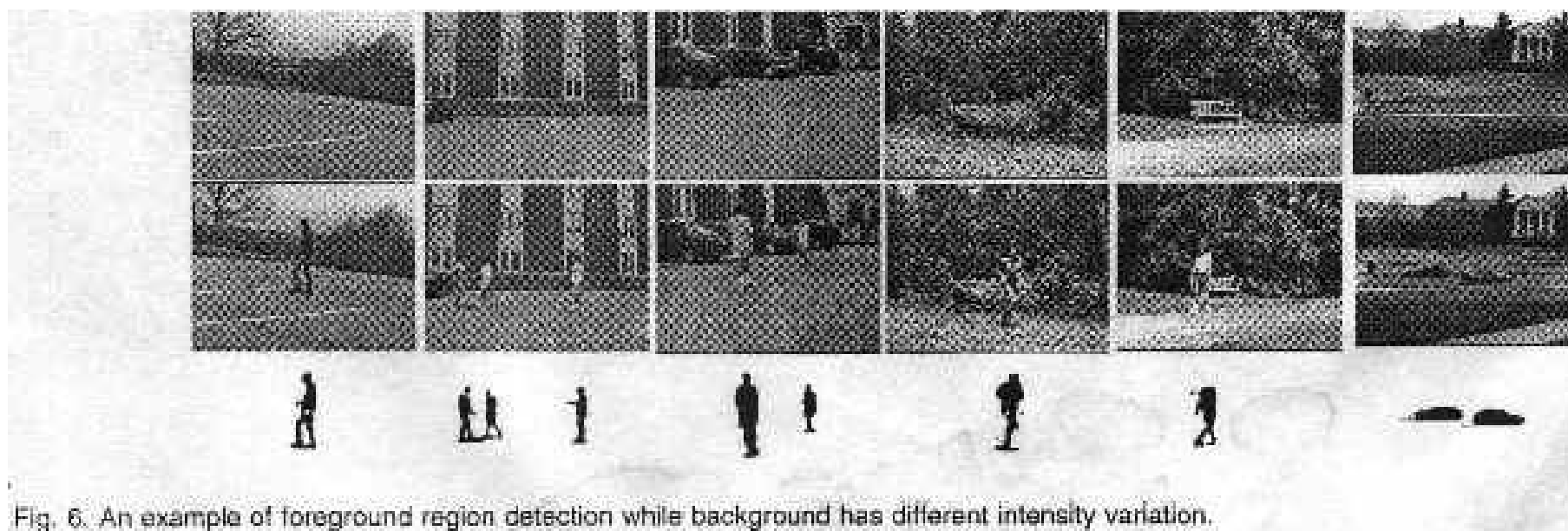|	k=2	k=3	k=4	K=6	k=8
Seq 1	0.89	0.72	0.72	0.51	0.32
Seq 2	0.52	0.40	0.29	0.03	0.02
Seq 3	0.81	0.70	0.59	0.41	0.28
Seq 4	0.85	0.77	0.70	0.50	0.40
Seq 5	0.85	0.77	0.71	0.59	0.46
Seq 6	0.87	0.77	0.72	0.60	0.48

Fig. 6. An example of foreground region detection while background has different intensity variation.

Consider Both Spatial and Temporal Domains

Application: Motion Furrow Detection

255   **Gray Values**   0

1st Frame

Edge/Line Detection in the Spatial Domain

$-$ $+$

$|\bullet| >$ **Threshold**
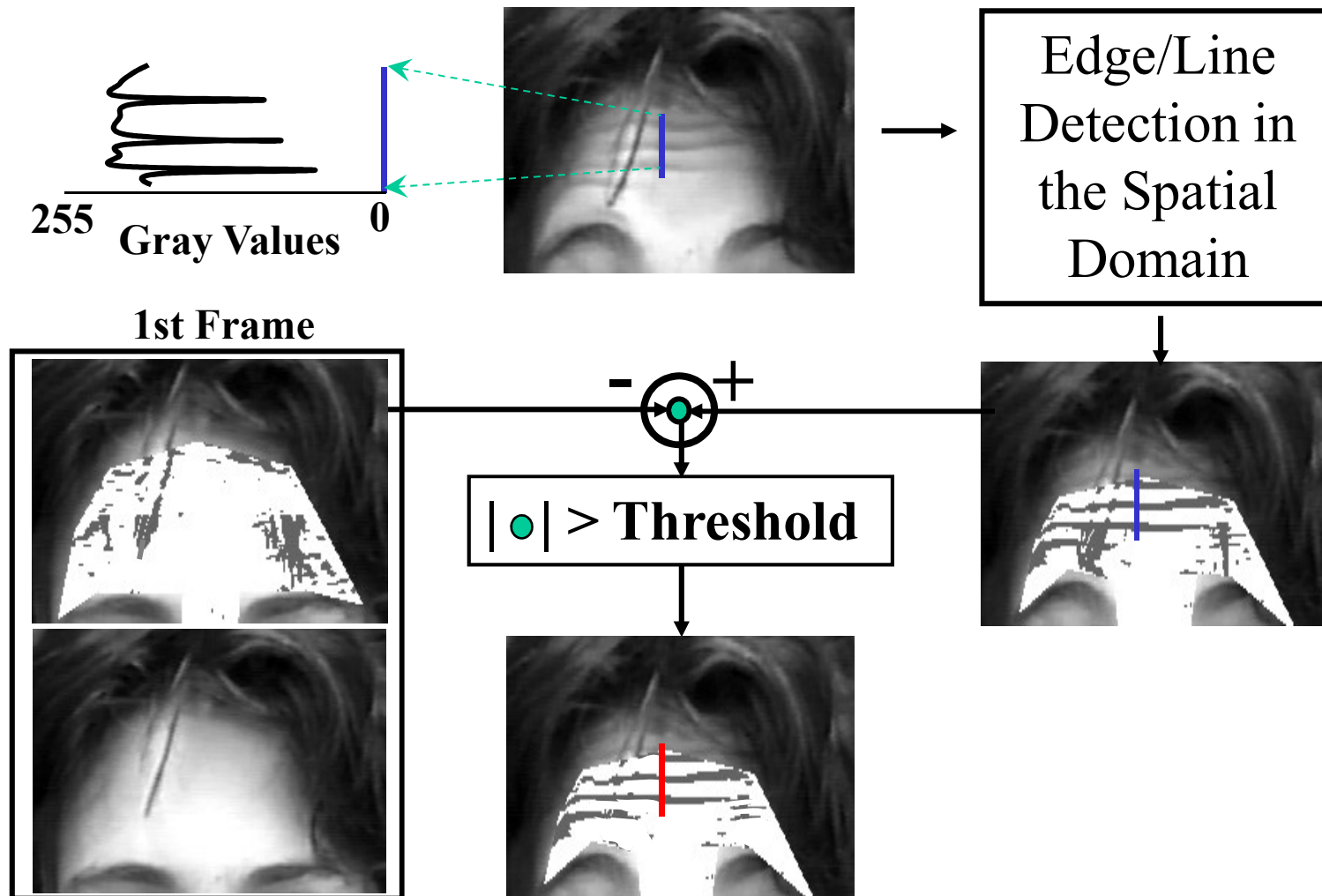
References

1. I. Haritaoglu, D. Harwood, and L.S. Davis, "W4: Real-Time Surveillance of People and Their Activities," IEEE PAMI, Vol. 22, No. 8, pp. 809-830, August 2000.

2. A. Elgammal, D. Harwood, L. Davis, "Non-parametric Model for Background Subtraction," 6th European Conference on Computer Vision, Dublin, Ireland, June/July 2000.