

Human motion analysis: Tools, models, algorithms and applications

Antoine Manzanera
ENSTA-ParisTech

August, 5, 2009

Tutorial presentation

Subject

Human motion understanding by analysis of video sequences

- Wide application spectrum.
- Rapid development domain.
- Numerous immature aspects.

Objectives

- Not an exhaustive review.
- Taxonomic approach.
- Focus on recent techniques.
- Introduction to technical tools.

Application fields

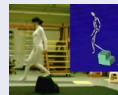
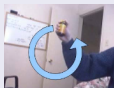
Smart videosurveillance

- 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



Problem variability

Those different applications may present extreme variability in terms of:

Scale

- Part(s) of the body
- Human bodies
- Group of humans

Acquisition conditions

- Controlled or not
- Mono vs Multi camera
- Stationary vs Moving camera

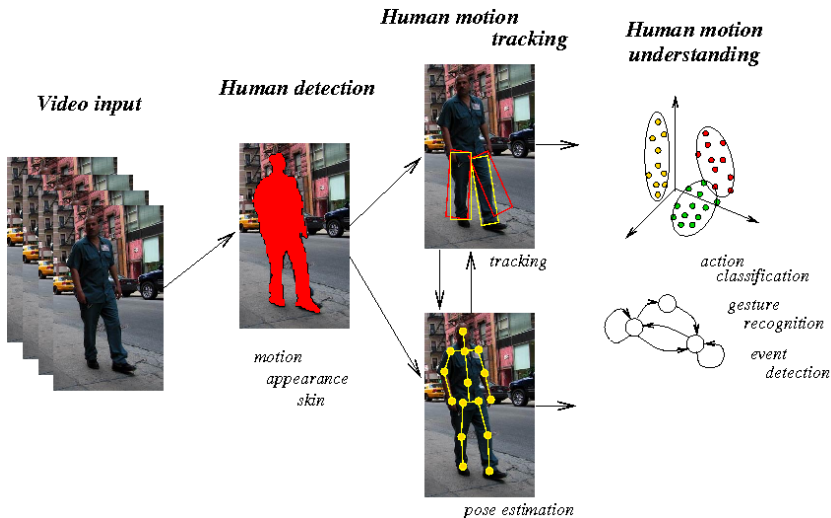
Parameters extraction

- Articulated pose estimation
- Activity understanding
- Qualitative action recognition

Tutorial scope

- No specific scale
- No particular application
- Acquisition not or few controlled (no markers or dedicated equipment)
- Monocular camera: 2d or monocular 3d
- Stationary camera: static background
- More focused on visual functions - Less on machine learning

Global framework



Outline

- 1 Human detection
 - Motion segmentation
 - Human silhouette
- 2 Tracking and model fitting
 - Tracking fundamentals
 - Model free tracking
 - Model based tracking
- 3 Human motion understanding
 - Visual features
 - Action recognition
- 4 Conclusion and Bibliography

Outline

- 1 Human detection
 - Motion segmentation
 - Human silhouette
- 2 Tracking and model fitting
 - Tracking fundamentals
 - Model free tracking
 - Model based tracking
- 3 Human motion understanding
 - Visual features
 - Action recognition
- 4 Conclusion and Bibliography

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



Background subtraction

Problem: What parameters represent within every pixel ?

Recursive exponential filter

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

If $|I_t - B_{t-1}| > T$, I_t is Foreground, else I_t is Background.

- α is the learning rate ; $\alpha \approx \frac{1}{t}$
- uses a fixed threshold T
- Only one parameter and one addition/shifting per pixel.

Two phased exponential filter

$$B_t = B_{t-1} + \alpha_1(I_t - B_{t-1}) ; \text{if } I_t \text{ is Background}$$

$$B_t = B_{t-1} + \alpha_2(I_t - B_{t-1}) ; \text{if } I_t \text{ is Foreground } (\alpha_2 \ll \alpha_1)$$

Temporal density estimation

A more general method consist in estimating for every pixel the temporal density function:

Temporal density definition

$$f_t(x) = P(I_t = x)$$

The density function can then be used to perform a statistically robust background estimation:

Density based background estimation

$$B_t = B_{t-1} + f_t(I_t)K(I_t - B_{t-1})$$

- K normalizing constant
- High cost in memory

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 $t > 0$:
 - $f_t(l_t) = f_{t-1}(l_t) + \varepsilon$
 - Renormalize f_t

The background B_t can also be defined as the *median* value, using $F_t^{-1}(1/2)$, where $F_t(i) = \sum_{j < i} f_t(j)$.

The temporal density can also be used directly for foreground segmentation: l_t is foreground if $f_t(l_t) < T$.

Gaussian background estimation

A good trade-off can be to consider a known distribution, such as the Gaussian model, which can be defined by using only two parameters $\{\mu, \sigma\}$, which are recursively estimated:

Recursive Gaussian estimation

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

If $|D_t| > n\sqrt{V_{t-1}}$, I_t is Foreground, else I_t is Background.

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

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

- M_t is the Gaussian mean, V_t is the Gaussian variance.
- n is an integer, typically 2 or 3.
- $\alpha_t = \alpha_1$ if I_t is background, $\alpha_t = \alpha_2$ otherwise ($\alpha_2 \ll \alpha_1$).

Σ - Δ background estimation

If the density f_t looks like a Zipf distribution, the density based estimation $B_t = B_{t-1} + f_t(I_t)KD_t$ looks like a Heaviside function:
 $B_t = B_{t-1} + \varepsilon$ if $D_t > 0$, and $B_t = B_{t-1} - \varepsilon$ if $D_t < 0$.

Σ - Δ background estimation

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

$$B_t = B_{t-1} + \text{sgn}(D_t)$$

$$V_t = V_{t-1} + \text{sgn}(n|D_t| - V_{t-1})$$

If $|D_t| > V_t$ I_t is Foreground, else I_t is Background.

- $\text{sgn}(x) = x/|x|$ if $|x| > 0$, 0 elsewhere.
- Full version: the update frequency of background B_t is proportional to the variance V_t .
- Extremely efficient from a computational point of view: 2 parameters, reduced instruction set, fixed point arithmetic.

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.

- N the number of modes, is typically between 3 and 7.
- W^i represent the weights of the different modes.

Silhouette based human detection

Certain systems perform the human detection using the shape of the silhouette output by the background subtraction step.

Example: [Kuno 96]

- Divide every circumscribing rectangle into 3 equal parts.
- Compute the smoothed projection histogram in every part.
- Every silhouette is described by 6 features:
 - The aspect ratio of the circumscribing rectangle.
 - The 3 normalized means of H1, H2, H3.
 - The 2 differences between the normalized standard deviations of H1 - H2 and H2 - H3.
- Compare the computed features with those of the typical patterns.

Typical patterns [Kuno 96]



Figure 3: Pattern A - crossing

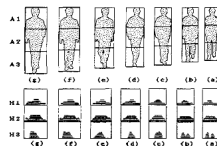


Figure 4: Pattern B - approaching

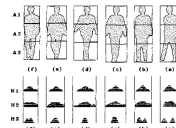


Figure 5: Pattern C - obliquely approaching

Shadow removal

Basically, the motion detection algorithms also detect the moving cast shadow, which is generally a problem.

On grayscale images, the problem is very difficult, since the cast shadows can only be detected using *a priori* on the geometry of the scene and objects, and on the illumination source position.

On color images, the problem can be addressed by performing the motion detection using an invariant color space (i.e. color features invariant to illumination changes). Such invariant are: normalized *rgb*, Hue *H*, Saturation *S*, or $c_1 c_2 c_3$:

$$c_1 = \arctan\left(\frac{R}{\max(G, B)}\right)$$

$$c_2 = \arctan\left(\frac{G}{\max(R, B)}\right)$$

$$c_3 = \arctan\left(\frac{B}{\max(R, G)}\right)$$

Example video [Salva 04]



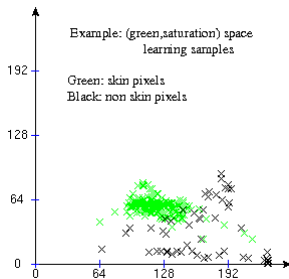
Skin detection

Skin segmentation is often a valuable clue for human close up color sequences. The most efficient reported techniques are based on Bayesian supervised classification in some features or color spaces.

Bayesian classification

- Sample a learning database and get the mean vector and covariance matrix for both skin (M_s, C_s) and non skin ($M_{\bar{s}}, C_{\bar{s}}$) pixel populations.
- Classify unknown pixels by minimizing the Mahalanobis distance:

$$c(x) = \arg \min_{k \in \{s, \bar{s}\}} {}^t(x - M_k) C_k^{-1} (x - M_k)$$



Human appearance

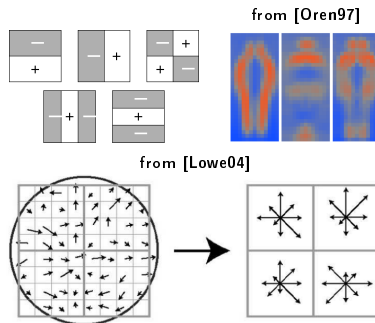
It is also possible to detect people from their *static* visual appearance. This has been done in particular in the case of *pedestrian detection*, where the camera is moving and the kinematic parameters hard to extract.

Visual features

- Over-complete Haar wavelet basis
- Histogram of Gradient descriptors

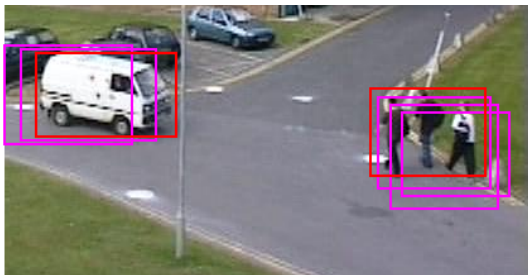
Learning Framework

- Support Vector Machine
- Cascaded weak classifiers



Motion based selection

On the contrary, the kinematic parameters that can be extracted from putting in correspondance the blobs in consecutive detection images (e.g. Velocity of the center of mass) can be useful to discriminate human from non human objects.



Outline

- 1 Human detection
 - Motion segmentation
 - Human silhouette
- 2 Tracking and model fitting
 - Tracking fundamentals
 - Model free tracking
 - Model based tracking
- 3 Human motion understanding
 - Visual features
 - Action recognition
- 4 Conclusion and Bibliography

Preliminaries

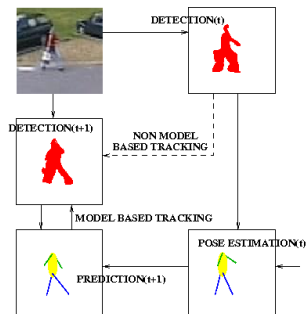
Tracking

Matching objects or object parts within consecutive video frames using visual features (points, lines, regions, ellipses,...)

Model fitting (Pose estimation)

Inferring the position of an articulated model from the visual observation

The two concepts are deeply related through the typical framework:



Fundamental tracking tools

Tracking can usually be expressed through the general framework:

$$\arg \max_{\theta \in S} P(\Theta_t = \theta / \Theta_{t-1} = \theta', X_t = x)$$

where S represent the *state* space; θ is a multi-dimensional state configuration, which can include (explicitely or implicitly) position, velocity, action category, etc, of the tracked objects; X_t represents the observation from the image space.

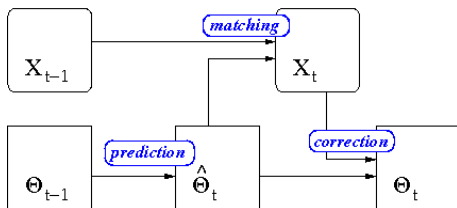


Image space similarity measures

Correlation

The normalized scalar product of the 2 centered sub-images f_t and f_{t-1} on a limited support $Supp(f)$ is a useful similarity measure, with values between -1 (perfect dissimilarity) and +1 (identity). Let \tilde{f} denote the mean value of f over $Supp(f)$:

$$\chi_t(f) = \frac{\sum_{x \in Supp(f)} (f_t(x) - \tilde{f}_t(x))(f_{t-1}(x) - \tilde{f}_{t-1}(x))}{\sqrt{\sum_{x \in Supp(f)} (f_t(x) - \tilde{f}_t(x))^2} \sqrt{\sum_{x \in Supp(f)} (f_{t-1}(x) - \tilde{f}_{t-1}(x))^2}}$$

Distribution similarity

The *Bhattacharyya* index, with values between 0 and 1 (perfect match) measures the similarity between 2 distributions. Let h be the normalized histogram of f on $Supp(f)$:

$$\mathcal{B}_t(h) = \sum_{j \in Supp(h)} \sqrt{h_t(j)h_{t-1}(j)}$$

State space estimation and prediction

Kalman Filter

State estimation technique performed through prediction and correction steps based on linear stochastic equation, optimal under the assumption of Gaussian distribution of the state space.

Condensation algorithm

State estimation technique based on sampling the state space posterior distribution from the visual observation, and iteratively propagating new samples from successive images.

Kalman Filter

The principle of the Kalman is to estimate the state $\Theta \in \mathbb{R}^n$ of the discrete time process governed by the linear stochastic equation:

$$\Theta_t = A\Theta_{t-1} + BU_t + W_{t-1}$$

using a measurement $X_t \in \mathbb{R}^m$ that relates to Θ as follows:

$$X_t = H\Theta_t + V_t$$

A is a $n \times n$ matrix relating two states at consecutive times.

B is an (opt.) $n \times l$ matrix related to a control input $U \in \mathbb{R}^l$.

H is a $m \times n$ matrix relating the state to the measurement.

V and W are ind., white Gaussian centered random vectors:

$$p(V) \simeq \mathcal{N}(0, R); p(W) \simeq \mathcal{N}(0, Q)$$

Kalman Filter algorithm - from [Welsh01]

Init. Start with initial estimates of $\hat{\Theta}_0$ and P_0 .

Then for $t > 0$:

Kalman Filter (1) Prediction phase

- 1 Project the state ahead

$$\hat{\Theta}_t^- = A\hat{\Theta}_{t-1} + BU_t$$

- 2 Project the error covariance ahead

$$P_t^- = AP_{t-1} {}^tA + Q$$

Kalman Filter (2) Correction phase

- 1 Compute the Kalman gain

$$K_t = P_t^- {}^tH(HP_t^- {}^tH + R)^{-1}$$

- 2 Update estimate with new measurement

$$\hat{\Theta}_t = \hat{\Theta}_t^- + K_t(X_t - H\hat{\Theta}_t^-)$$

- 3 Update the error covariance

$$P_t = (I - K_tH)P_t^-$$

Condensation algorithm

Condensation algorithm [Isard98]

Iterate:

input: $\{s_{t-1}^{(n)}, \pi_{t-1}^{(n)}, c_{t-1}^{(n)}\}_{n \leq N}$ the set of N old samples

output $\{s_t^{(n)}, \pi_t^{(n)}, c_t^{(n)}\}_{n \leq N}$ the set of N new samples

1 **Select** a sample $\hat{s}_t^{(n)}$ as follows:

- select (uniformly) a random number $r \in [0, 1]$
- find the smallest j for which $c_{t-1}^{(j)} \leq r$
- set $\hat{s}_t^{(n)} = s_{t-1}^{(j)}$

2 **Predict** the new sample $s_t^{(n)}$:

$$s_t^{(n)} = \arg \max P(\hat{\Theta}_t = s / \Theta_{t-1} = \hat{s}_t^{(n)})$$

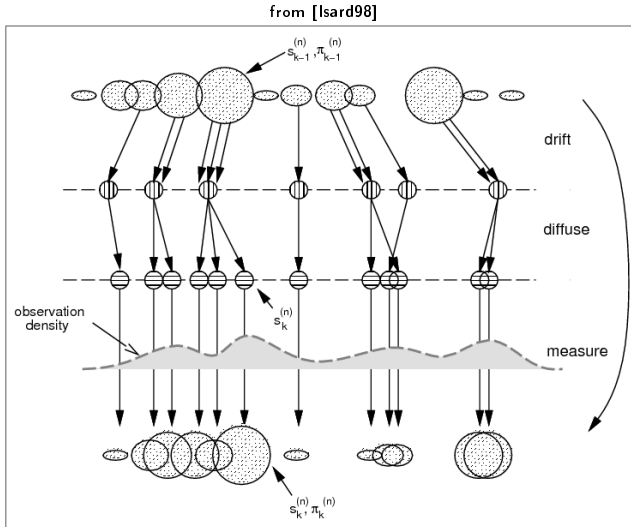
3 **Correct** from the measure and update the current weight:

$$\pi_t^{(n)} = P(X_t / \Theta_t = s_t^{(n)}), \text{ normalizing so that } \sum_{i \leq N} \pi_t^{(i)} = 1$$

then recompute the cumulative distribution:

$$c_t^{(0)} = 0; c_t^{(n)} = c_t^{(n-1)} + \pi_t^{(n)} (1 \leq n \leq N)$$

Condensation algorithm, cont.

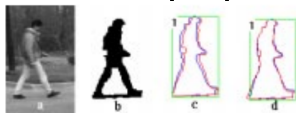


Tracking the body as a whole ($W^4 - 1$)

In the W^4 [Hari98] system, the first part of the human body tracking consists in the following phases:

- The *centroid* of each object of the detection output is computed as the *median* coordinate of every connected component (silhouette). (less sensitive to the moving extremities than the center of mass).
- For every tracked object, the new centroid position is estimated using a second order motion model (constant acceleration, needs 3 points to be computed).
- From this initial estimate, the final position is obtained by maximizing the binary correlation between the shifted previous silhouette and the actual silhouette.

W^4 - from [Hari98]

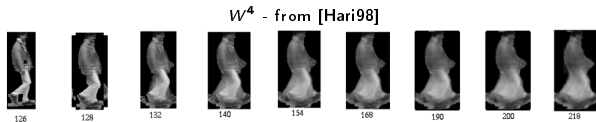


Tracking the body as a whole ($W^4 - 1$) cont'

- To deal with disambiguation in the case of merging and splitting silhouettes, the system also perform gray level correlation, using *temporal texture template* defined as follows:

$$\Psi_t(x, y) = \frac{I_t(x, y) + \omega_{t-1}(x, y) \times \Psi_{t-1}(x, y)}{\omega_{t-1}(x, y) + 1}$$

(Coordinates (x, y) are relative to the centroid; $I_t(x, y)$ correspond to the foreground value computed at time t ; the weight $\omega_t(x, y)$ counts the number of times coordinate (x, y) has been detected as foreground). Such template favors most confident body features such as head or torso.



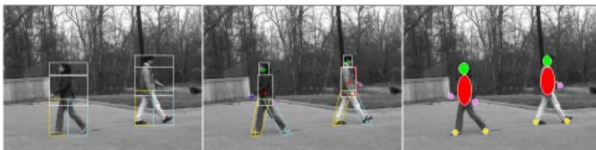
Tracking body parts ($W^4 - 2$)

In the W^4 [Hari98] system, the 2d part of the human body tracking is based the *cardboard* model based tracking:

- Every bounding box (BB) is splitted into head, torso and legs BBs according to constant morphometric proportions (left image)
- The width of those 5 BBs are computed using the *median* width of the silhouettes within their initial BB, and the first order *moments* of the silhouettes are used to compute the *principal axes* of the BB, which represent the initial guess for the body pose estimation (center image)
- The feet and legs position are also initially estimated using extreme points of the lower and torso parts respectively (right image)
- Starting from this prediction, the position of the different parts of the bodies are refined using again the temporal texture template:

$$\Psi_t(x, y) = \frac{I_t(x, y) + \omega_{t-1}(x, y) \times \Psi_{t-1}(x, y)}{\omega_{t-1}(x, y) + 1}$$

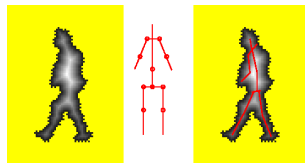
W^4 - from [Hari98]



Pose estimation

One important tool for fitting a model over a 2d silhouette is the distance transform computed over the silhouette;

- The distance transform associates to every pixel of the foreground its distance to the background.
- The fitness function associated to a given pose of the 3d model can be defined by the sum of the distance transform function over the pixels of the projected model.
- The best pose (in the correction phase) is the one which maximizes the fitness function.



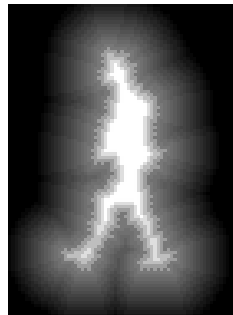
However: (1) The high dimension of the space configuration is an important problem for the optimization, and (2) The 2d silhouette is by nature ambiguous as it may correspond to different 3d poses.

Dimensionality reduction [Elga04]

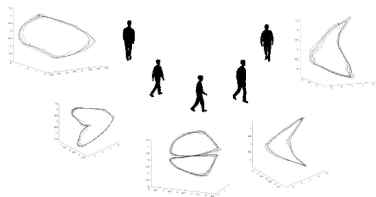
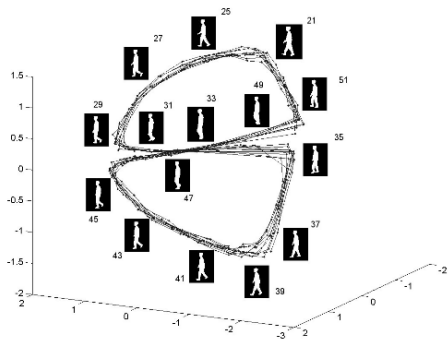
Using an intermediate representation space of low dimension improves the pose inference efficiency, and reduces the effect of ambiguities.

Manifold learning [Elga04]

- The silhouette is represented using signed distance transform (right).
- For each orientation view, the silhouettes are embedded into a low dimensional space using *Locally Linear Embedding* framework which preserves the neighbourhood relations of the original space while reducing dramatically the dimensionality using local linearity constraints (next page - left).
- The different views are learned separately through different manifolds (next page - right).



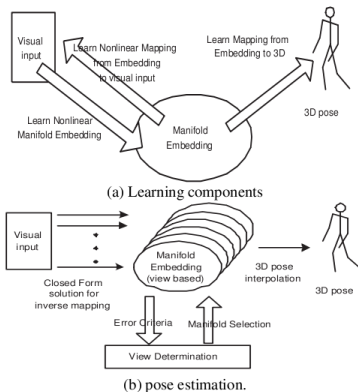
Dimensionality reduction [Elga04] cont.



3d pose estimation [Elga04]

Mapping learning and pose inferring [Elga04]

- Learning mapping from embedded manifold to the visual space representation is performed using *Generalized Radial Basis Function* (GRBF), with constrains guaranteeing the invertibility of the mapping.
- Likewise, mapping from embedded manifold to 3d pose space is made with GRBF.
- No state space tracking or temporal information is explicitly used in this framework.



Dimensionality reduction [Jaeg09] - 1

In the work of [Jaeg09], dimensionality reduction is done both in 3d pose and visual spaces. Furthermore, state space - including activity - tracking is performed.

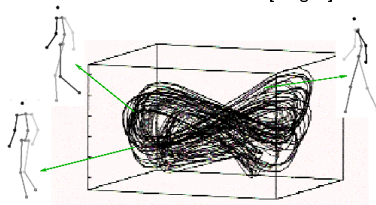
Pose Manifold Embedding [Jaeg09]

- *Locally Linear Embedding* is used to project the 20 dimension learned samples of 3d model pose within a 4d manifold.
- Explicit reconstruction in the original pose space is learned using a kernel regressor and *Relevance Vector Machine* (RVM):

$$X = W_p(\Phi_p(x))$$

where x is a 4-dimension vector, Φ_p 4 dimension Gaussian kernel, W_p is a sparse 20×4 matrix, X a 20-dimension vector.

3d view of the 4d manifold, from [Jaeg09]



Dimensionality reduction [Jaeg09] - 2

In the image space, the appearance of the foreground silhouette is also represented in a low dimension space using *Principal component analysis* (PCA):

Appearance descriptor [Jaeg09]

The appearance model is reduced to approx. 10 dimension components using:

- Binary PCA applied on the binary silhouette.
- PCA applied on the distance transform of the silhouette.



Body pose tracking [Jaeg09] - 1

Dynamic prior [Jaeg09]

For each action category (run, walk, . . .), the pose prior is modelled combining dynamic prior learned from RVM and static prior learned as *Gaussian Mixture Model*:

$$p(x_t/x_{t-1}) \simeq p_d(x_t/x_{t-1})p_s(x_t)^{1/\lambda}$$

Appearance prediction and Pose inference [Jaeg09]

- Appearance in the low dimensional space y can be predicted through the generative model from the pose in the manifold space x and the body orientation ω using Bayesian sampling and RVM:

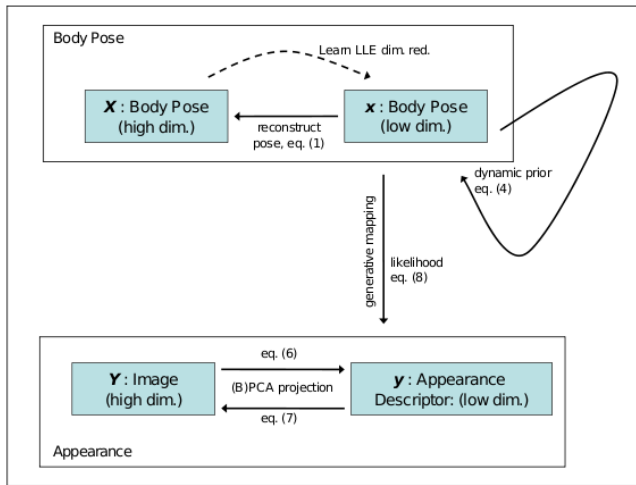
$$p(y/x, \omega) \mathcal{N}(y; W_a \Phi_a(x, \omega), \Sigma_a)$$

- State space estimation is performed using a variant of the condensation algorithm on a state space combining the low dimensional pose x with other parameters: bounding box positions p and dimensions d , body orientation ω , and activity category a .

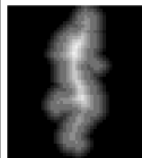
$$\Theta_t = \{x_t, p_t, d_t, \omega_t, a_t\}$$

- Note that the system is also able to perform *activity recognition*, but the complexity dramatically increases with the number of activity categories.

Body pose tracking [Jaeg09] - 2



(b)



(c)

(a)

Outline

- 1 Human detection
 - Motion segmentation
 - Human silhouette
- 2 Tracking and model fitting
 - Tracking fundamentals
 - Model free tracking
 - Model based tracking
- 3 **Human motion understanding**
 - Visual features
 - Action recognition
- 4 Conclusion and Bibliography

Human motion understanding

Objectives

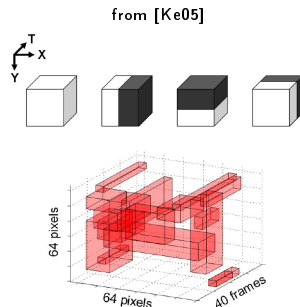
This step must provide *symbolic* outputs with *high level semantics* about the observed human action.

Contexts

- Gesture Recognition:
Visual interface, Sign Language Recognition, ...
- Action / Event Categorization:
Run, Hand clap, Sit, Stand up, ...
- Unusual Motion Detection:
 - Video surveillance : Aggression, Distress, ...
 - Bio medical applications : Gait pathologies, ...

Haar volumetric features

The 2d visual features used to detect human appearance can be extended to 3d to recognize human action. For example, in [Ke05], 3d spatio-temporal Haar features are applied on the apparent motion field (optical flow), and then used within a learned cascade of binary classifiers.



Motion History Images

Some techniques aim at representing the action by a single image, for recognizing it using classical shape matching methods. For example the *Motion History Images* (MHI) are introduced in [Bob96]:

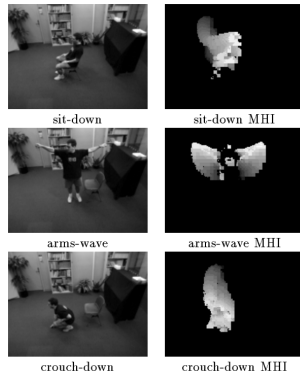
$$H_{\tau}(x, y, t) = \begin{cases} \tau & \text{if } D(x, y, t) = 1 \\ \max(0, H_{\tau}(x, y, t - 1) - 1) & \text{otherwise} \end{cases}$$

where τ is the time depth, D is the motion segmentation label.

Action recognition is then performed using the 7 first *Hu moments* of the MHI images. Those shape descriptors ensure translation and scale invariance.

Then *Mahalanobis distance* is used with respect to some learned action prototypes.

from [Bob96]



Spatial Motion Histograms

To reduce the dimension of the action visual space, [Zhong04] compute a spatial motion histogram by performing motion segmentation (center), then reducing the spatial resolution and counting the number of times a moving object is detected on every square region, within a certain time depth (right). Vector quantization is then applied to reduce the action visual descriptors to K prototypes using K -means algorithm.

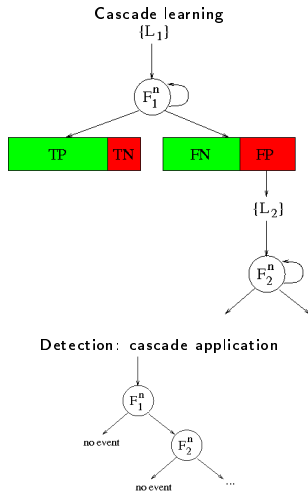
from [Zhong04]



Filter cascades

The rare event detection framework is used by [Ke05] to learn the weak classifier cascade using volumetric Haar filters:

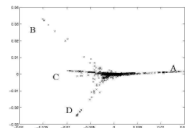
- From an initial learning examples, a first set of Haar filters is learned by iteratively selecting a collection of features, until a certain detection and false alarm rate is reached.
- The first weak classifier then correspond to a majority voting of all the filters.
- The next node of the cascade is then trained the same way from a learning set extracted from the false positive examples.



Unusual cluster detection

In [Zhong04], a technique directly inspired by text document clustering using keyword occurrence is used to detect unusual events. Taking as text document the video sequence, and as keywords the prototypes of spatial motion histograms obtained from the learned examples, detecting unusual events correspond to identifying the isolated clusters in a low dimensional embedded co-occurrence space:

from [Zhong04]



Outline

- 1 Human detection
 - Motion segmentation
 - Human silhouette
- 2 Tracking and model fitting
 - Tracking fundamentals
 - Model free tracking
 - Model based tracking
- 3 Human motion understanding
 - Visual features
 - Action recognition
- 4 Conclusion and Bibliography

Conclusion

Many efforts remain to be done to fill the semantic gap:

- The higher level must take into account sophisticated adaptation techniques to be more robust to segmentation or location errors from the low level functions.
- The lower level must also combine more sophisticated models to deal with the most difficult cases (e.g. radial motion, occlusion, crossing, . . .)

Conclusion, cont.

The most promising systems combine an impressive bunch of - often recent - techniques from different fields:

Background modelling

- Gaussian
- Σ - Δ
- Generalized distribution

Image descriptors

- Corner points
- Wavelets
- Color invariants
- Distance transforms

Image space tracking

- Cross correlation, centroid, ...
- Transformed spaces
- Density kernel based

State space tracking

- Kalman filter
- Condensation

Dimensionality reduction

- PCA, BPCA, ...
- LLE, IsoMap, ...

Learning frameworks

- Bayes, GRBF, ...
- SVM, RVM, ...
- Cascade, Boosting, ...

Bibliography - Reviews

 **[Agga99]** J.K. AGGARWAL and Q. CAI

Human motion analysis: a review

Computer Vision and Image Understanding 73(3), 428-440. (1999)

 **[Wang03]** L. WANG, W. HU and T. TAN

Recent Developments in Human Motion Analysis




Pattern Recognition 36(3), 585-601. (2003)

 **[Poppe07]** R. POPPE




Vision-based human motion analysis: An overview

Computer Vision and Image Understanding 108(4), 4-18. (2007)

Bibliography - Background subtraction

-  **[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)
-  **[Manza07]** A. MANZANERA and J. RICHEFEU
A new motion detection algorithm based on Sigma-Delta
background estimation.
Pattern Recognition Letters 28(3), 320-328. (2007)

Bibliography - Human features

-  **[Kuno96]** Y. KUNO et al
Automated detection of human for visual surveillance system
Proc. of ICPR 865-869. (1996)
-  **[Oren97]** M. OREN et al
Pedestrian Detection Using Wavelet Templates
Proc. of CVPR 193-199. (1997)
-  **[Salva04]** E. SALVADOR, A. CAVALLARO and T. EBRAHIMI
Cast shadow segmentation using invariant color features
Computer Vision and Image Understanding 95, 238-259. (2004)

Bibliography - Tracking fundamentals



[Isard98] M. ISARD and M. BLAKE

CONDENSATION - CONDitional DENsity propagATION for visual tracking

Int. Journal of Computer Vision (1998) 29(1), 5-28 (1998)






[Welsh01] G. WELSH and G. BISHOP

An Introduction to the Kalman Filter

Tutorial of ACM SIGGRAPH (2001)

Bibliography - Tracking human

-  **[Hari98]** I. HARITAOGLU, D. HARWOOD and L.S. DAVIS
W⁴: Who? When? Where? What? A Real Time System for
Detecting and Tracking People Automated detection of human for
visual surveillance system
Proc. of Int. Conf. on Face and Gesture Recognition (1998)
-  **[Elga04]** A. ELGAMMAL and C-S. LEE
Inferring 3D body pose from silhouettes using activity manifold
learning
Proc. of CVPR Vol. 2, 681-688. (2004)
-  **[Jaeg09]** T. JAEGLI, E. KOLLER-MEIER and L. VAN GOOL
Learning generative models for multi-activity body pose estimation
Int. Journal of Computer Vision (2009) 83, 121-134 (2009)

Bibliography - Understanding human motion

 **[Bob96]** A.F. BOBICK and J.W. DAVIS

Real-time recognition of activity using temporal templates
Proc. of Workshop on Applications of Computer Vision 39-42.
(1996)

 **[Zhong04]** H. ZHONG, J. SHI and M. VISONTAI

Detecting unusual activity in video
Proc. of CVPR Vol. 2, 819-826. (2004)

 **[Ke05]** Y. KE, R. SUKTHANKAR and M. HEBERT

Efficient visual event detection using volumetric features
Proc. of ICCV Vol.1, 166-173. (2005)